DISEASE PROGRESSION MODELING USING MULTI-DIMENSIONAL CONTINUOUS-TIME HIDDEN MARKOV MODEL

A Thesis
Presented to
The Academic Faculty

by

Yu-Ying Liu

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Interactive Computing

Georgia Institute of Technology
December 2015

DISEASE PROGRESSION MODELING USING MULTI-DIMENSIONAL CONTINUOUS-TIME HIDDEN MARKOV MODEL

Approved by:

Professor James M. Rehg, Advisor
School of Interactive Computing
Georgia Institute of Technology

Professor Jimeng Sun
School of Computational Science and Engineering
Georgia Institute of Technology

Professor Aaron Bobick
School of Interactive Computing
Georgia Institute of Technology

Professor Hiroshi Ishikawa
School of Medicine
University of Pittsburgh

Professor Irfan Essa
School of Interactive Computing
Georgia Institute of Technology

Date Approved: 21 August 2015

To the people that have faith in me.

ACKNOWLEDGEMENTS

First I would like to express my sincere gratitude to my advisor, James Rehg. His broad knowledge, passion in research, unique insights, and warm care to students establish a role model of scholars to me.

I am also grateful to my committee and several researchers that I have worked with. I would like to thank Hiroshi Ishikawa, Gadi Wollstein, and Joel Schuman for instructing me doing research in Ophthalmology, Fuxin Li, Le Song, and Shuang Li, for their great help in formulating and solving the core problems in my thesis, Aaron Bobick, Irfan Essa, and Jimeng Sun for their insightful comments for my thesis, and Mei Chen for her advice on my initial research projects.

Many thanks to all my colleagues who always give me spiritual supports. Special thanks to Ping Wang, Howard Chou, Jianxin Wu, David Tsai, Fuxin Li, and Amrita Gupta, for their warm care and company.

Finally I would like to thank my family for their endless support, especially to my husband, Yong-Dian Jian, and my mother, Mi-Yun Lo, whose love and encouragement gives me the strengths and wisdom to move forward through the challenges. The last but not the least, I would like to thank my daughter Jasmine and son Henry, whose arrival to this world gives a new meaning to my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The goal of this thesis is to develop a general tool for disease progression modeling which can handle arbitrary observation times, support a multi-dimensional (M-D) view of progression as the co-evolution of multiple interacting biomarkers, capture complex nonlinear patterns, and provide visualizations that effectively communicate the dynamics of disease to domain experts. The development of an M-D view of disease progression is important since many diseases are characterized by several temporally-evolving processes in structure, function, and biochemical, whose interaction can reveal critical but previously undiscovered mechanism.

The Continuous-time hidden Markov model (CT-HMM) is a useful tool in modeling disease progression based on noisy observations of disease states which arrive at irregular sample times. Unfortunately, the modeling flexibility provided by the CT-HMM comes at the cost of a more complex inference procedure than the standard discrete-time HMM. There is no widely-accepted algorithm for efficient parameter learning in the existing literature, and numerical optimization is often utilized directly in maximum likelihood estimation. This has restricted the use of CT-HMM to small models or to applications that make restrictive assumptions on the state transition timing.

In this thesis, we propose to use CT-HMM with M-D gridded state structure to model disease progression, where each dimension represents the change of one or a set of disease markers. By learning dynamic interactions from longitudinal datasets, progression trajectories and patterns can be explored and visualized in the full spectrum of disease evolution, leading to greater insights into population level behavior,

enabling prediction of future progression, and the identification of phenotypes. The development of M-D CT-HMM models for large state spaces requires the development of efficient parameter learning methods.

We present the first complete characterization of Expectation-Maximization (EM)-based learning methods in CT-HMM, which both extends and unifies prior work on continuous-time Markov chain (CTMC) models. We address two technical challenges: the estimation of posterior state probabilities, and the computation of end-state conditioned statistics in a CTMC. We efficiently discretize the estimation of posterior state probabilities into a discrete time-inhomogeneous HMM, and present soft and hard EM algorithms. The benefits and drawbacks of these developed methods are analyzed and experimentally validated relative to the existing literature. For CT-HMM hidden path decoding, we present a novel method for computing the path-conditioned expected state duration, which are useful in trajectory decoding and clustering tasks.

Our M-D CT-HMM methods are evaluated on three real-world datasets from Glaucoma, Alzheimer's disease, and Hypertension, with applications including visualizations of the progression, prediction of future measurements, and trajectory clustering. The visualization results of disease progression provide a novel insight into the global structure of progression, and the prediction results outperform the state-of-the-art method for Glaucoma. Our promising results demonstrate that M-D CT-HMM paired with visualizations can provide insights into disease evolution and support the development of individualized treatment plans, resulting in cost-effective disease management.

CHAPTER I

INTRODUCTION

Thesis Statement Continuous-time hidden Markov model with multi-dimensional state structure can model disease progression as the co-evolution of multiple biomarkers and support visualizations of complex dynamics, leading to greater insights into population level behavior, enabling prediction of future progression and the identification of phenotypes.

The goal of disease progression modeling is to learn a model for the temporal evolution of a disease from longitudinal clinical measurements obtained from a sample of patients. By distilling population data into a compact representation, disease progression models can yield insights into the disease process through the visualization and analysis of disease progression trajectories (see Fig. 1,2 for an example). In addition, disease models can be utilized to predict the future course of disease in an individual, supporting the development of individualized treatment schedules with improved efficiencies. Progression models can also support phenotyping by providing a similarity measure between trajectories for grouping patients based on their progression.

Hidden Markov model (HMM) is an attractive choice for modeling disease progression for three reasons: (1) it supports the abstraction of a disease state which is useful to model staged progression with complex dynamics; (2) it can deal with noisy measurements effectively by using data emission models from states and leveraging population transition behavior; (3) it can easily incorporate dynamical priors and model structural constraints. The conventional discrete-time HMMs have been used to model disease progression [91], but they are not suitable in general because they assume that measurement data is sampled regularly at discrete intervals and

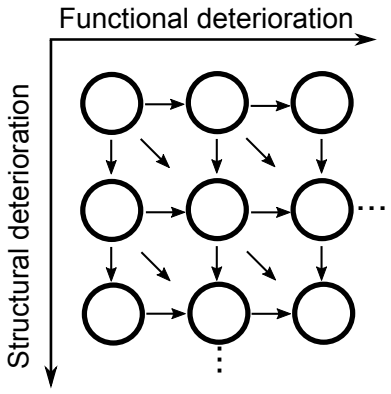


Figure 1: A 2-dimensional state structure for Glaucoma progression modeling, for capturing the interaction between structural and functional degeneration. The arrows represent the allowed *instantaneous* transitions.

state transitions can only happen at these discrete times. In reality patient visits are frequently *irregular* in time, as a consequence of scheduling issues, missed visits, and changes in symptomatology, and the underlying disease process also evolves in continuous, arbitrary time.

A *Continuous-time* HMM (CT-HMM) is an HMM in which the transitions between hidden states can occur at arbitrary continuous times, and the observational data can also arrive irregularly in time [16] . It is therefore more suitable for modeling disease process and the clinical measurements. In this dissertation, we propose to use CT-HMM with multi-dimensional (M-D) gridded state structure (an example 2-D model is in Fig. 1) to model disease progression, where each dimension represents the change of one or a set of biomarkers. By using M-D gridded disease states defined by M kinds of disease progression aspects (such as structural, functional, biochemical, etc.) and learning their dynamic interactions over time from longitudinal datasets, population-level progression trajectories can be explored in the full spectrum of disease evolution. Furthermore, the learned model can be utilized to predict future progression trajectory and find potential progressing phenotypes when combined with clustering techniques.

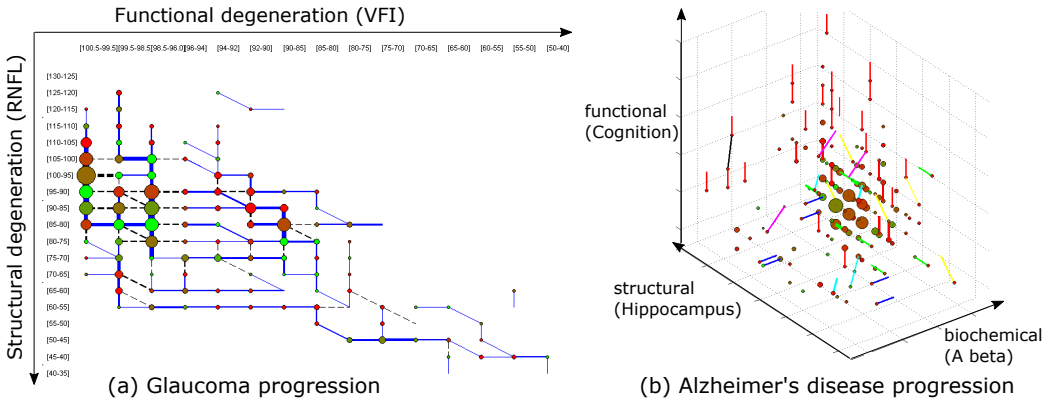


Figure 2: Visualization scheme: (a) Glaucoma disease progression: the strongest transition among the three instantaneous links from each state are shown in blue while other transitions are drawn in dotted black. The line width and the node size reflect the expected count. The node color represents the average sojourn time (red to green: 0 to 5 years and above). (b) Alzheimern's disease progression: the vis scheme is similar to (a) but the strongest transition link from each state is color coded as follows: $A\beta$ direction only (blue), *hippo* only (green), *cog* only (red), $A\beta + hippo$ (cyan), $A\beta + cog$ (magenta), $hippo + cog$ (yellow), $A\beta + hippo + cog(black)$. The node color represents the average sojourn time (red to green: 0 to 3 years and above).

One motivating application of our MD CT-HMMs is for modeling Glaucoma progression, which is characterized by two temporally-evolving processes: structural and functional degeneration, and their relationship is clinically-found to be complex and non-linear. There are conflicting findings in the temporal ordering of detectable structural and functional degeneration, which confound glaucoma clinical assessment and treatment plans [97], resulting in inefficient treatment. We use a 2D state structure where one dimension is functional deterioration, and the other is structural degeneration, for modeling Glaucoma progression (Fig. 1). Our model can flexibly capture the nonlinear relationship between composite disease states, revealing subtle dynamics that the conventional longitudinal model, such as linear regression, cannot represent. Our learned 2D CT-HMM from longitudinal dataset can not only show whether structural degeneration precedes functional loss in the early disease stages, but can also reveal at what structural ranges the onset of functional degeneration happen more frequently (Fig. 2(a)).

Our M-D disease progression models bridges the gap between the simple 1-D staging models (mild to severe) usually used in clinics and the complex disease-specific models often designed in research. The proposed M-D models generalize the conventional 1-D staging models with more expressive power. For clinical use, the M-D models with validated predictive power can be utilized to plan a cost-effective visiting interval and to inform treatments. In research side, the results from our general-purpose models can be used to generate hypotheses of the disease mechanism, and aid the preliminary design of the more sophisticated disease-specific models. Our M-D models paired with visualization function acts as a general exploratory tool for studying the dynamics of a disease evolution from multiple biomarkers, complementing other existing disease models.

We now turn to discuss the technical challenges in learning CT-HMM. Though CT-HMM is a more suitable model than DT-HMM for disease modeling, unfortunately the additional modeling flexibility provided by the CT-HMM comes at the cost of a more complex inference procedure. In CT-HMM, not only are the hidden states unobserved, but also their *transition times* are unobserved. It is also possible for multiple unobserved hidden state transitions to occur between two successive observations. An early parameter estimation algorithm circumvents these challenges in decoding hidden information by directly maximizing the data likelihood [29] using standard numerical optimization. However, the general numerical methods, which require calculation of derivatives of likelihood function, are not efficient enough for learning large-scale CT-HMM models (we will show a complexity analysis shortly). The lack of efficient learning algorithms in the literature has limited CT-HMM to applications of small models [29] or for large-scale models requires unrealistic assumptions on state transition timing [42, 52]. There is a need for learning methods which can scale to large state space sizes.

With the challenges of learning CT-HMM, one may resort to simply discretize

the time horizon and just use discrete-time HMM. However, the time horizon for state changes in medical conditions can vary dramatically. In early stages of disease a state change might not occur for years, but in an acute phase they could occur very frequently. For states with very short expected dwelling times, the discrete time step needs to be sufficiently small. However, this might be inefficient for dealing with changes that occur once several years. On the other hand, if the discretization is too coarse, many transitions could be collapsed into a single one, obscuring the real dynamics in continuous-time. In contrast, CT model performs inference over arbitrary timescales using a single matrix exponential. We believe CT is a better choice than DT for modeling continuous-time processes such as disease progression and clinical data.

To tackle the challenges of CT-HMM learning, if we consider a simpler case where the hidden states can be observed directly without noise but only at irregularly-sampled times, then the model is equivalent to a continuous-time Markov chain (CTMC) model with incomplete observed data. Expectation-Maximization (EM) algorithms for maximum likelihood estimation (MLE) in CTMC with incomplete observation was developed in [64]. It works by computing expected state durations and state transition counts [7, 25] conditioned on each pair of successive state observations in the E-step, and use these calculated sufficient statistics to update parameters in the M-step.

In computing these conditioned expectations in CTMC, [64] finds that efficient closed-form evaluations are obtained when the transition intensity matrix can be diagonalized through an eigendecomposition (abbreviated as *Eigen*). Such an approach has been extended to CT-HMM [90] recently. However, the intensity matrix is frequently not diagonalizable during learning, and this approach often fails in real data [64]. More recently, [25] demonstrates that the necessary conditional expectations can also be computed using two alternative and more general approaches: *matrix*

exponential (abbreviated as *Expm*) and *uniformization* (abbreviated as *Unif*). Unfortunately, none of these two methods have been explored or developed for CT-HMM, which has the ability to correctly account for measurement noise and offers more flexibility to clinicians in describing the relationship between measurements and the true disease states.

In this dissertations, we present the first comprehensive framework for efficient parameter learning in CT-HMM models, which both extends and unifies prior work on CTMC models. Based on the finite number of observations, our framework discretizes the estimation of posterior state probabilities at observation times into a time-inhomogeneous hidden Markov model, and then incorporates two approaches to estimating the hidden state distributions for the corresponding observations. In the "hard" approach, the distribution over hidden states is approximated by the single most likely (Viterbi) decoding, which results in significant computational savings. The "soft" approach, in contrast, maintains the full distribution over the hidden states, gaining accuracy at the cost of increased computation. The benefits and drawbacks of these approaches combined with the three different approaches (*Expm, Unif, Eigen*) for computing the end-state conditional expectations [25] are analyzed and validated experimentally both by simulation and by running on real-world disease longitudinal datasets. We find that soft EM approach achieves higher accuracy and is more robust to noises than hard EM. *Expm* in computing conditioned expectations works best in soft EM while *Unif* can achieve the best efficiency in hard EM due to its decomposability.

In addition to developing novel EM algorithms for CT-HMM parameter learning, we also review and discuss the state sequence decoding problem in CT-HMM. The decoding of the hidden states corresponding to the actual observations can be solved using *Viterbi* algorithm by forming a time-inhomogenous embedded Markov chain. However, the problem of solving the most probable state sequence in between two

specified states with a total time, is unsolved until recently [46, 66]. Furthermore, the problem of finding the *globally* most probable state sequence given a set of irregularly observed measurements seems to still be an open question as there is no prior literature. In this dissertation, we focus on exploring the simpler problems of finding the optimal state sequence given *two (decoded) end-states* and a total time. It is discovered recently in [66] that the problem of finding the best state sequence marginaling out the state dwelling time is always well-defined while the solution of the most likely state and duration trajectory does not always exist when the model parameters have a found property [66].

For the well-defined problem of finding the optimal state sequence marginaling out the dwell time given two end-states and a total time, [46] proposes a path extending and pruning method which finds the exact best state sequence in finite time. We augment this approach in also finding the expected state duration given the optimal path and the total time, which we name it as computation of *path-conditioned* expected state duration. We show that the three methods (*Eigen, Expm, Unif*) that computes the end-state conditioned statistics, can also be used here by working on an auxiliary matrix constructed for the path. We also derive a new closed-form based on Laplace transform to compute these durations with better time complexity than the above methods but with the prerequisite that the holding time parameters are distinct. We compare the time complexity of all methods and gives simulation results. The developed decoding methods can be utilized to understand the most probable hidden transition dynamics and in trajectory clustering tasks.

In application domains, we utilize a 2-D CT-HMM for Glaucoma progression modeling in structural and functional dimension. Our results supports the finding in [95] that retinal nerve fiber layer (RNFL) thickness of around 77 microns acts as a tipping point at which functional deterioration became clinically observable with structural deterioration. In addition to qualitative visualization, we devise a simple

procedure for predicting future continuous measurements using the learned model. Our 2D CT-HMM gives better prediction results than the state-of-the art method using Bayesian joint linear regression [62] for Glaucoma, which shows the practical value of CT-HMM for effective disease prognosis and management.

Besides conducting detailed analysis for Glaucoma progression modeling, we also do a preliminary study for Alzheimer's disease using 3-D CT-HMM with structural, functional, and biochemical markers, and a 2-D blood pressure longitudinal analysis for Hypertension using a 2-D CT-HMM on Electronic Health Record (EHR) data. Our preliminary results in Alzheimer's disease corroborates the recent finding that Amyloid β (Aβ) is an early marker before detectable cognition decline and structural loss [17]. For hypertension, our trajectory analysis finds two distinct progression clusters and their visualization clearly show the different transition trends, which demonstrate the potential use of our model for finding progression phenotypes.

This dissertation is organized as follows. Chapter 2 reviews the related state-based disease progression models and compare them to our M-D CT-HMM. Chapter 3 presents the proposed framework and novel EM algorithms for CT-HMM parameter learning with experimental results from simulations and on real-world disease datasets. Chapter 4 reviews the literature for decoding the hidden optimal state sequence given two end-states and a total time, and then presents our methods for computing path-conditioned expected state durations with analysis and simulation results. Chapter 5 gives detailed analysis and prediction results for Glaucoma progression modeling, and Chapter 6 shows preliminary studies for Alzheimer's disease and Hypertension. Finally Chapter 7 gives conclusion and discusses future directions.

Table 1: Time complexity comparison of all presented methods in evaluating the required expectations under Soft/Hard EM for CT-HMM learning (r: number of distinct time interval, S: number of states, L: number of edges, V: number of visits, M: the truncation point for *Unif*, set as $\lceil 4 + 6\sqrt{\hat{q}t_\Delta} + (\hat{q}t_\Delta)\rceil$, where $\hat{q} = max_i\ q_i$).

complexity	Expm	Unif	Eigen
Soft EM	$O(r(2S)^4 + rL(2S)^3)$	$O(\hat{M}S^3 + rS^3M^2 + rS^2LM^2)$	$O(rS^5 + rLS^4)$
Hard EM	$O(r(2S)^4 + rL(2S)^3)$	$O(\hat{M}S^3 + \min(rS^2, V)SM^2$ $+ \min(rS^2, V)LM^2)$	$O(\min(rS^2, V)S^3$ $+ \min(rS^2, V)LS^2$

1.1 Overview of main contributions

- Introduce CT-HMM with multi-dimensional (M-D) state structures for disease progression modeling, which can statistically characterize the co-evolution multiple biomarkers, revealing the dynamic interactions in a more informed picture of disease progression.

- Pair the M-D CT-HMM with novel state-based visualizations, facilitating comprehension of disease progression patterns.

- Present the first comprehensive framework for CT-HMM parameter learning using EM, which both extends and unifies prior work on CTMC models.

- Develop several variants of EM algorithms for CT-HMM learning, including *(Soft,Hard)* $\times$ *(Expm,Unif,Eigen)* combinations. The time complexity (see Table 1) and accuracy of the developed methods are analyzed and compared experimentally by simulation and on real-world datasets. Our results show that soft EM approach achieves higher accuracy and is more robust to noises than hard EM. *Expm* in computing conditioned expectations works best in soft EM while *Unif* can achieve the best efficiency in hard EM due to its decomposability. Our work enables more efficient learning of CT-HMM than the prior method which uses direct numerical optimization (see Table 2 for a complexity comparison).

Table 2: Time complexity comparison for CT-HMM learning using a direct numerical optimization method (use Broyden-Fletcher-Goldfarb-Shanno (BFGS) [29]) to find MLE and the proposed EM methods in this thesis. For BFGS, LVS^2 represents the evaluating of the likelihood function using Forward-Backward algorithm for each of L parameters when calculate numerical differences. rS^3 represents the precomputation of matrix exponential for r distinct time intervals, which also needs to do for each of L parameters when compute numerical difference. L^2 costs is in updating the approximated Hessian. In practice, V, the number of total visits in a dataset, can be very large. Direct numerical optimization can be very time consuming due to the LVS^2 term.

complexity	Direct numerical opt.	End-state conditioned EM
CT-HMM	BFGS [29]: $O(LVS^2 + LrS^3 + L^2)$	Our thesis work Soft(Expm): $O(VS^2 + r(2S)^4 + rL(2S)^3)$ Hard(Unif): $O(VS^2 + \hat{M}S^3 + \min(rS^2, V)SM^2 + \min(rS^2, V)LM^2)$

- For CTMC an CT-HMM decoding tasks, derive new closed-forms for computing total time conditioned state sequence probability and path-and-total time-conditioned expected state durations, which has better time complexity than other alternative methods.

- Demonstrate applications of M-D CT-HMM on several disease contexts: Glaucoma, Alzheimers disease, and Hypertension. Insights of the diseases are gained by visualizing and analyzing the learned state transition tendencies. For Glaucoma, the results on predicting future measurements outperforms the state-of-the-art Bayesian joint linear regression method.

CHAPTER II

RELATED WORK ON STATE-BASED DISEASE PROGRESSION MODELING

In this chapter, we review several related work using state-based disease progression models. These include: *CT-HMM models* which handle both a continuous-time state transition process and noisy observation process (include a small CT-HMM in Sec. 2.1.1, a hybrid CT-HMM in Sec. 2.1.2, a large-scale CT-HMM modeled as a time-inhomogeneous HMM in Sec. 2.1.3), *other CT type models* (include CTMC which observes the state directly without noise handling [29], continuous-time latent Markov model for phase-type duration modeling using 2-layers of CTMCs in Sec. 2.2.1, and continuous-time Bayesian network (CTBN) in Sec. 2.2.2), and *non CT models* (include discrete-time HMM in Sec. 2.3.1, Kalman filtering in Sec. 2.3.2, and Bayesian probability model for event ordering in Sec. 2.3.3). We also briefly review other widely-used non-state based longitudinal models in Biostatistics in Sec. 2.4.

In each section, we will briefly explain each referred disease model, and compare the referenced model to our multi-dimensional CT-HMM , in terms of their ability to model the same disease and dataset, and their possible pro and cons.

For state-based disease progression modeling, we believe that continuous-time models are better than the discrete-time alternatives, such as DT HMM and Kalman filter. Using these discrete-time methods to model continuous-time process, such as disease evolution and clinical data, have both efficiency and accuracy issues. Note that the time horizon for state changes in medical conditions can vary dramatically. In early stages of disease a state change might not occur for years, but in an acute phase they could occur very frequently. The intervals between clinical visits can also

widely-vary depends on the disease conditions among patients. For states with very short expected dwelling times, the discrete time step needs to be sufficiently small. However, this might be inefficient for dealing with changes that occur once several years. On the other hand, if the discretization is too coarse, many transitions could be collapsed into a single one, obscuring the real dynamics in continuous-time. In contrast, CT model performs inference over arbitrary timescales using a single matrix exponential. We believe CT is a better choice than DT for modeling continuous-time processes such as disease progression and clinical data.

2.1 Continuous-time hidden Markov model

2.1.1 Small CT-HMM

In [29], the paper reviews the prior works that used CTMC and CT-HMM for disease progression modeling with panel data (observations at arbitrary times).[1] The prior applications include (1) screening for abdominal aortic aneurysms [26], for breast cancer [13], and for cervical cancer [37] ; (2) problems following lung/heart transplantation [27],[75],[38]; (3) staging for hepatic cancer [34], [6]; (4) complications for diabetes [56]; (5) HIV infection and AIDS [72].

The commonly used disease models in the literature are shown in Fig. 3 for staged progression (example in Fig. 5) and Fig. 4, the illness-death model (examples in Fig. 6). The model in Fig. 3 consists of a series of successively more severe disease states, and an absorbing state (dealth). The patient may advance into or recover from adjacent disease states, or directly jump to the death state. The stages of disease may be modelled as a continuous-time (hidden) Markov process, with a time-homogeneous or time- inhomogeneous transition rate matrix.

This paper introduced an **R** package, **msm**, that allows a multi-state model to be fitted to longitudinal data. Features of msm include the ability to learn transition

[1] In biostatistics, the term **multi-state model** is used more often to describe a process (*Markov* or not) in which an individual moves through a series of states in continuous time.

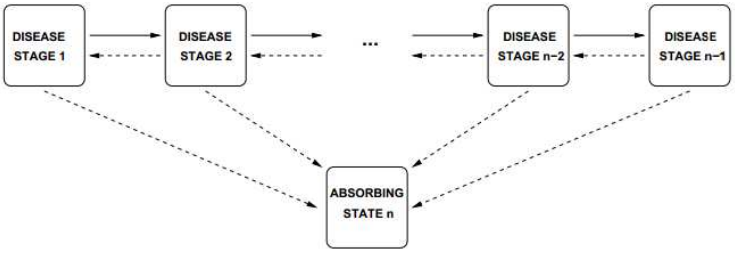


Figure 3: A conventional disease progression model which has a linear state chain. At each state, multi-dimensional features may be used, but the detailed temporal relationship among these features can not be captured in this linear chain model (from [29], used without permission).

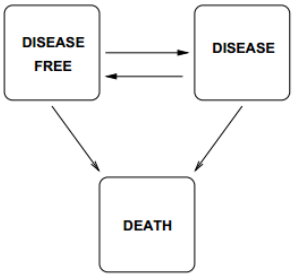


Figure 4: Healthy-illness-death model (from [29], used without permission).

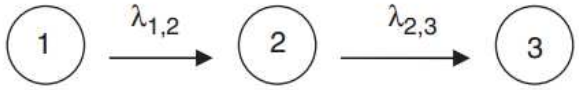


Figure 5: A three-state model for breast cancer progression. State 1: no detectable breast cancer; state 2: preclinical cancer, detectable by screening (asymptomatic); state 3: clinical breast cancer (symptomatic) (from [88], used without permission).

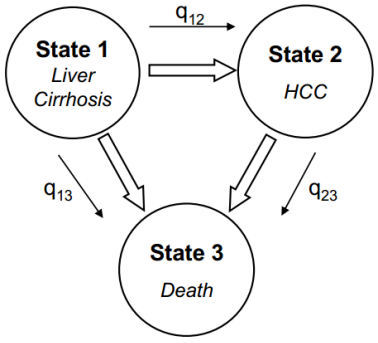


Figure 6: A three-state hidden Markov model for modeling progressing from liver cirrhosis to hepatocellular carcinoma (HCC) (from [6], used without permission).

rates in CTMC and CT-HMM with/without covariates, and the ability to model data with a variety of observation schemes, such as censored states. However, in this package, direct numerical optimization methods, such as BFGS, are adopted for maximum likelihood estimation of model parameters. As we pointed out in Chapter 1, the direct numerical optimization method does not scale well with large state space.

Our M-D disease progression modeling generalizes the existing 1-D model, which can capture the dynamics among multiple factors. In addition, our proposed EM learning method can efficiently learn the parameters for a larger scale models of hundreds of states.

2.1.2 Hybrid CT-HMM

In [90], the paper proposes a probabilistic disease progression model which has three layers (see Fig. 7): the top layer is a CT-HMM for modeling the hidden progression stages, the middle layer is a set of comorbidities[2] whose onsets trigger the state transition of the top layer, and the bottom layer are the observed clinical findings, which are activated by the hidden comorbidities. A bipartite noisy-or discrete-time Bayesian Network is used to model the relationship between the middle comorbidity layer and the bottom observed clinical finding layer (see Fig. 8). The model is demonstrated on Chronic Obstructive Pulmonary Disease (COPD) patient cohort with Electronic Health Record (EHR) data, and some interesting clinical insights are gained.

Aiming to learn a general-purpose model for any chronic disease based on a general input data type, such as EHR, the authors do not assume prior knowledge of either the ground truth progression stages or the key indicators that signify the stage transitions. Since the onset of a new comorbidity often signifies the exacerbation of the target disease, the authors use the onset pattern of multiple comorbidities to collectively

[2]A comorbidity is a disease or syndrome that co-occurs with the target disease. For example, hypertension is a common comorbidity of diabetes and osteoporosis is a common comorbidity of COPD [90].

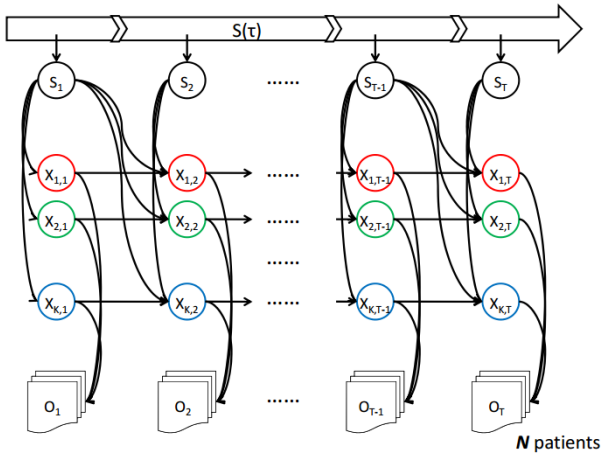


Figure 7: A disease model of three layers of variables: S are hidden progression state variables, X are hidden comorbidity variables, and O are observed clinical findings (from [90], used without permission).

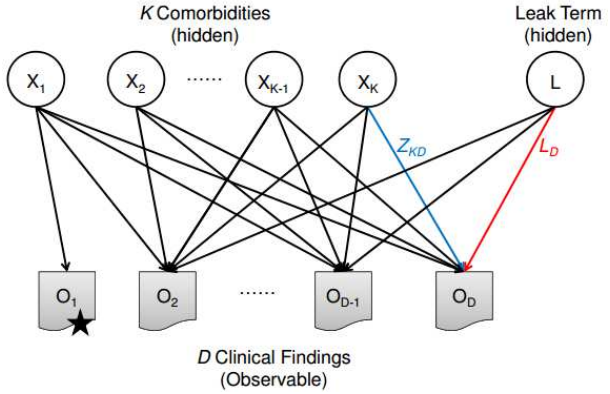


Figure 8: The noisy-or Bayesian network. The clinical findings can be activated by a hidden comorbidity or by the always-on hidden cause (the leak term). The starred finding O_1 is an anchor, which means it can only be activated by a specific comorbidity (X_1 in this example). (from [90], used without permission).

drive the state transitions of the target disease. Also, it is assumed that once a comorbidity is turned on, it can not be turned off in the future.

An EM learning algorithm with Gibbs sampling to jointly model the distribution of hidden variables and model parameters is used for parameter estimation. For COPD disease, 6 progression states and 10 comorbidities are used, where each comorbidity is guided with a set of predefined anchor observations to improve interpretability. The results are in Fig. 9 and Fig. 10. In Fig. 9, one can see that the prevalence of

some comorbidities raise rapidly with the state transitions, while some comorbidities remain more stable. In Fig. 10, the inference result of two patients, one has a stable and another has a progressive trajectory are shown, which reveal the heterogeneity of different progression patterns.

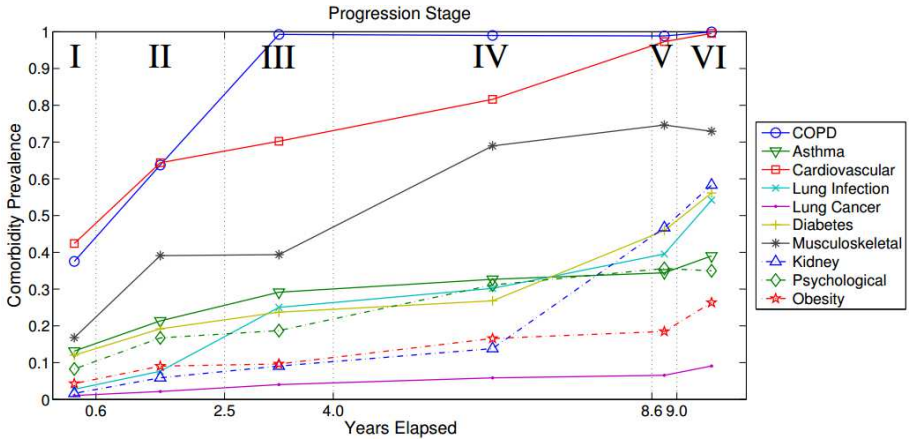
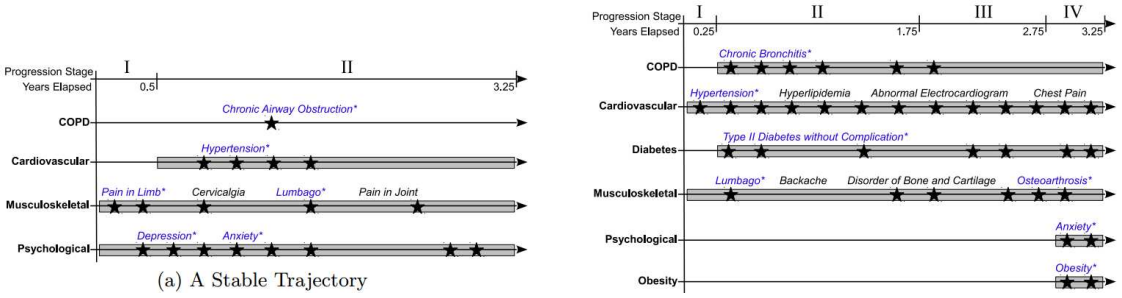


Figure 9: The progression state over time versus the comorbidity prevalence averaged over 10000 generated virtual patients (from [90], used without permission).



(a) A Stable Trajectory

(a) A stable trajectory

(b) a progressive trajectory

Figure 10: Inference of the progression trajectory and comorbidity onset of two patients. Inferenced stages and comorbidity onsets are shown in the top and the gray bar respectively. The stars denote the observed clinical findings (From [90], used without permission).

When compare this comorbidity-driven progression model to our MD CT-HMM model, our MD model is more suitable to directly model the interactions of the known key disease indicators. The other clinical observations as well as treatment effects can be incorporated as covariates, whose weights are to be learned from the training set.

The learned weights reveal whether a covariate is significant in influencing the state transition rates. And by using covariate model, each patient can have individualized state transition matrix at each visit. In addition, these covariates can be binary or continuous values, and they can be on or off at any visit time.

In contrast, the comorbidity model uses the onset patterns of hidden comorbidities to define the progression of the target disease, which complements existing standards that directly use the key indicators to define stages, in offering a comprehensive characterization of the patient's conditions. One restrictive assumption is that the comorbidity can not be turned off once it is on. We think that the new progression definition based on hidden comorbidities are novel. However, some hidden causes are not specific to the target diseases, but are common to many diseases, such as anxiety in psychological group and obesity, or it may be more related to aging. We think that some discriminative analysis and feature selection steps are required for this comorbidity model.

2.1.3 Large-scale CT-HMM with state transition constraints

In [42], the authors points out that the lack of efficient parameter estimation algorithm for CT-HMM has limited its applications to very small models. They thus develop a Baum-Welch like EM algorithm for large-scale CT-HMM, with restrictive assumptions that state transitions take place exactly and only at observation times. This assumption also implies that two states which has a path should also have an instantaneous link. We find that these assumptions may be too restrictive and not realistic for real-world applications.

The method is applied to analyze the temporal interactions among 74 psychiatric diseases, from around 375000 patients traced over 30 years with a total of around 7,000,000 clinical visits. The paper presents the visualization of the learned disease interaction (Fig. 11 and 12) so that possible causal interactions may be identified.

The prediction of future disorders given a current disease is also shown (see Fig. 13). It shows that the predicted future diseases (left) using the model are in accordance with the real trajectories (right) directly calculated from the data.

To our knowledge, this paper is the first application of CT-HMM with a large number of states, but an approximate parameter estimation algorithm with restrictive state transition assumption is used. Our new CT-HMM learning methods do not have these assumptions and can be applied to this application as well.

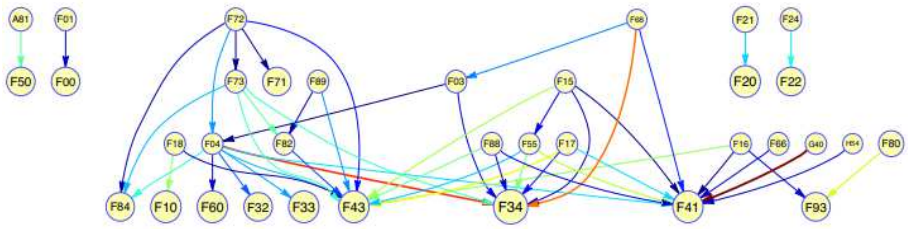


Figure 11: Visualization of the strongest interactions of the transition intensity q_{ij} (from [42], used without permission).

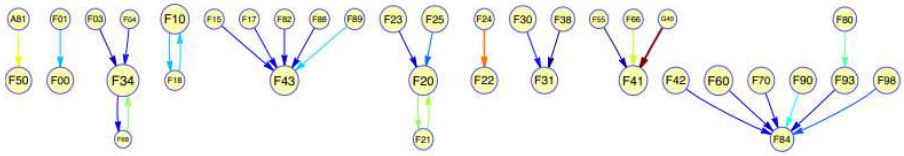


Figure 12: Visualization of the strongest interaction from the transition probability $p_{ij} = q_{ij}/q_i$ (from [42], used without permission).

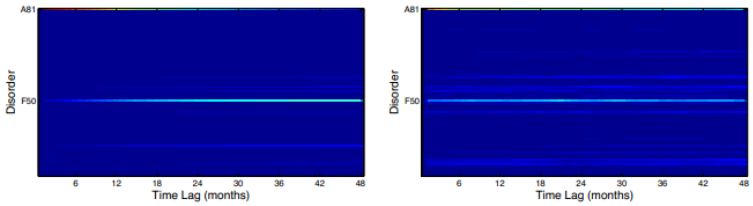


Figure 13: Prediction of future disorders (y axis) over time (x axis) when given a fixed current disease state (A81) (from [42], used without permission).

2.2 Other continuous-time models

2.2.1 Continuous-time latent Markov Model

A latent CTMC model is proposed in [86] for for more flexible, duration-dependent disease state sojourn time modeling than conventional CTMC, where sojourn time distribution is exponential and can be unrealistic for real applications. This model assumes that the disease process is characterized by an underlying latent CTMC, with multiple latent states mapping to each observed disease state (see Fig. 14). This model allows flexible sojourn time modeling while retain analytic tractability of data likelihood due to the CTMC framework. In contrast, the semi-Markov models that permit generic sojourn time distribution but yield intractable likelihood particularly if the model has reversible transitions.

In more detail, a latent CTMC structure implies phase-type (PH) distributions of sojourn times in each disease state. PH distributions are attractive in their ability to approximate generic distribution with positive support, and PH functionals are

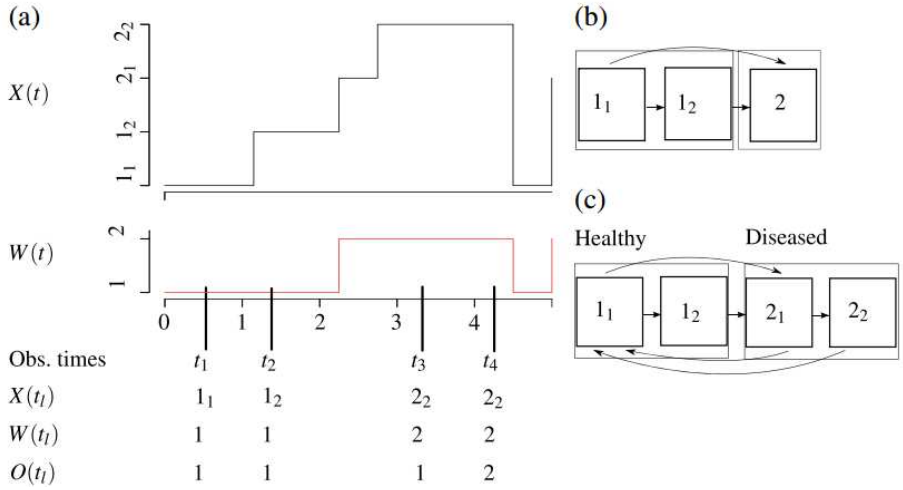


Figure 14: (a) An example of latent trajectory $X(t_l)$, disease trajectory $W(t_l)$, and observed data $O(t_l)$ at irregularly observed times for model structure in subfigure c, assuming possible observed error. (b) A two-state survival model assuming disease states $= 1, 2$ and latent states $= 1_1, 1_2, 2$, where state 2 is absorbing. (c) A two-state reversible model with disease states = 1=Healthy, 2=Diseased, and latent states $= 1_1, 1_2, 2_1, 2_2$. (from [41], used without permission).

expressible with matrix exponentials [41][86]. One disadvantage of PH is that model parameters may not be identifiable [41]. The latent CTMC models for disease processes inherit these advantages and disadvantages of PH distributions. In practice, models with few latent states are more likely to result in identifiable parameters.

The original inventor of latent CTMC [86] suggests using standard numerical optimization methods for obtaining MLE model parameters. However, this method is slow and exhibit poor convergence properties as noted in [41]. [41] proposes a new EM algorithm for latent CTMC that combines recent computational development derived for PH models [4] and discretely-observed CTMCs [7], which results in better time efficiency in learning.

Simulation study is conducted [41] for latent CTMCs using simple two-state survival and healthy-illness model, where each disease state has an underlying CTMC of 1 or 2 states. The interpretive power of latent CTMC models for describing disease process for a bronchiolitis obliterans syndrome (BOS) dataset of lung transplant patients. Overall, the results suggest mixed performance of latent CTMC estimates. The latent CTMC though flexible, are parametric and is subject to model misspecification.

As very small disease models are demonstrated in [41], we think that further investigation is needed to test the feasibility of latent CTMC to applications of a larger number of disease states with duration modeling.

2.2.2 Continuous-time Bayesian network

In [20], a continuous-time Bayesian network (CTBN) is utilized to diagnose cardiogenic heart failure and anticipate its likely evolution. This paper describes the first clinical application of CTBN, which overcomes the regular-sampling assumptions of dynamic Bayesian network (DBN). The model structure for diagnosing heart failure

and complications is shown in Fig. 15, which consists of both unobservable physiological variables and clinically observable events. By conducting inference on this model, one can predict the occurrence of complications (such as myocardial infarcation, shock) and the persistence in time based on the gathered evidence. The inference results are shown to be consistent with the current medical understanding.

The key parameters of the CTBN model, i.e., the conditional intensity matrix (CIM), are elicited on the basis of the medical expertise [20]. The authors provide a procedure to specify CIM by first entering the conditional probability table (CPT) that within a short-enough time interval, to represent the influence of the parent nodes on each node as the corresponding CIMs would do in continuous time. Please see [20] for more detail on parameter specification based on expert knowledge. In the future work, the authors would like to directly learn the CIMs from clinical data, which is challenging and is still an active research problem.

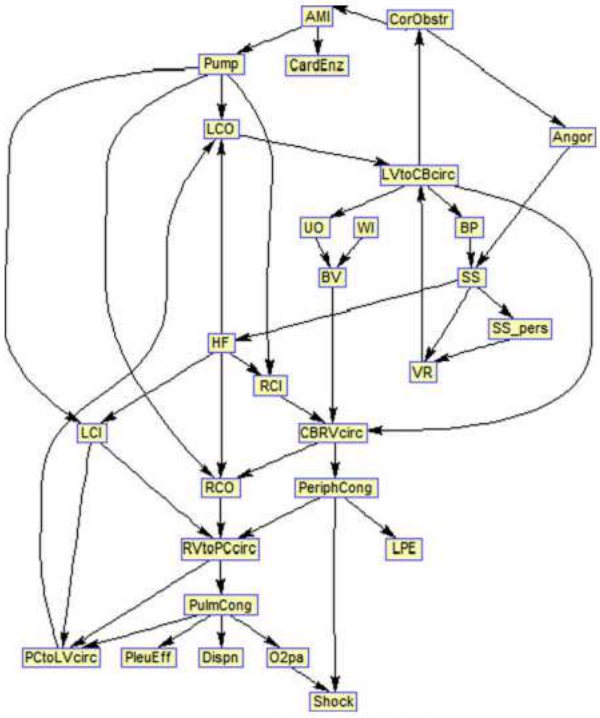


Figure 15: CTBN model for diagnosing cardiogenic heart failure (from [20], used without permission).

2.3 Non continuous-time models

2.3.1 Discrete-time HMM

In [91], the paper uses DT-HMM to model brain aging using 4D (3D + time) brain MRI images. A 5-state left-to-right DT-HMM, as shown in Fig. 16 is used to capture the changes of aging-related brain regions over time. In the results, the authors compared the patient age distribution at each state and the number of state transitions for two patient groups based on their cognition scores. One group is cognitive decline (CD), and the other is non-cognitive decline (NCD), classified based on the change slope of cognition scores. The main findings are as follows. For CD group, the mean age between two successive states are shorter (Fig. 17), and the frequency of state transitions are higher than the NCD group (Fig. 18).

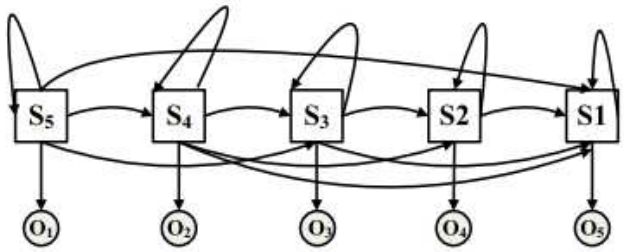


Figure 16: A 5-state left-to-right discrete-time HMM for modeling brain aging (from [91], used without permission).

State / Average Age	State 1	State 2	State 3	State 4	State 5
Relatively healthy group	**86.73**	**77.81**	70.54	67.11	65.96
Progressive group	83.34	73.59	71.67	71.16	69.44

Figure 17: Average age at each state for relatively healthy (Non-Cognition Decline) and progressive group (Cognition Decline) (from [91], used without permission).

We believe that our multi-dimensional CT-HMM progression model can be utilized in this application to analyze and visualize the interaction between structural degeneration (physical brain changes) and functional deterioration (cognition decline) directly. In more detail, the same left-to-right model can be trained first using the

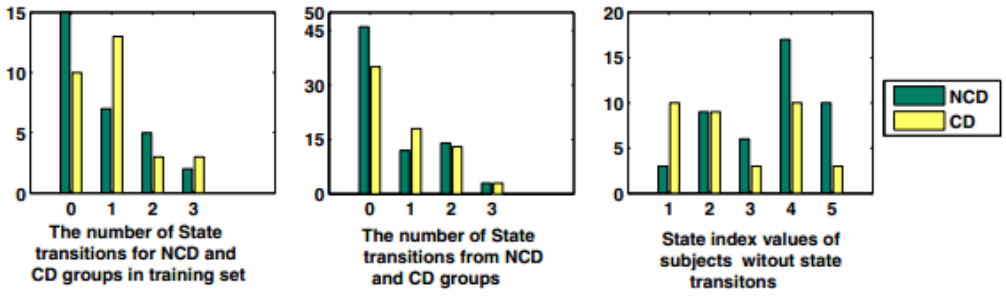


Figure 18: Histogram of number of state transitions from each state in NCD (non cognition decline) and CD (cognition decline) group (from [91], used without permission).

structural features, and the data emission model for each state is learned. Then the 2D state space is defined by bands of cognition scores and the learned structural states. This can be an interesting future work.

2.3.2 Kalman filter

In [73], Kalman filter is used to estimate the true value of biomarkers associated with Open-Angle Glaucoma (OAG) from noisy observations and the filtered values are then fed into a logistic regression function to classify patients as progressors or non-progressors. It is shown that using Kalman filter first resulted in data input which improved the classification accuracy compared to the model fed by raw measurements.

We compare the main differences between Kalman filter and CT-HMM model for data denoising. The state space in Kalman filter is continuous while that for HMM is discrete. One strong assumptions in Kalmain filter is that all hidden and observed variables have Gaussian distribution, while in HMM there is no specific constraints in data distribution. In addition, the basic Kalman filter is a linear system that has a global transformation matrix not varies with the current state, while in HMM the transition behavior depends on the current states. Regarding to the decoding step, Kalman filter is an on-line sequential processing method, while HMM uses Viterbi decoding to find the best hidden state sequence in a batch mode.

In modeling disease progression where non-linear transitions are expected over

time and among markers, we believe that state-dependent transition behavior are required to capture the state-varying progressing behavior. Thus, abstraction of disease states to a finite set may be needed. Furthermore, in settings where we are given a set of measurements from past visits, it is better to find the optimal hidden state sequence considering all these measurements, rather than using the sequential, on-line decoding method. Furthermore, we would like to use a continuous-time model where the observations can be irregularly spaced in time, while the basic Kalman filter requires regular discrete-time observations. Due to these reasons, we believe that CT-HMM is a more suitable choice in modeling and decoding the non-linear state-dependent transition behavior.

2.3.3 Bayesian probability model for event ordering

In [18], the paper introduces a new characterization of disease progression, which describes the disease as a series of events (we illustrate this in Fig. 19), and each event represents a significant change in some aspect. The aim is to learn the most common event ordering from measurements over a patient cohort. The applications to familial Alzheimer's and Huntington's diseases are demonstrated. The found event ordering model can be used to do patient staging. Also, more understanding about different but similar neurodegenerative diseases can be gained, by comparing their event orders respectively.

In the formulation, [18] treats longitudinal data as cross-sectional data, which means that visits from the same patients are treated as they are irrelevant. The data likelihood for a given event ordering S is formulated as follows. If cross-sectional data j is at position k in the event-based progression model, events $E_{s(1)}, ..., E_{s(k)}$ has occurred, while events $E_{s(k+1)}, ..., E_{s(N)}$ has not, the likelihood for that patient' data given event ordering S is: $p(X_j|S,k) = \prod_{i=1}^{k} p(x_{s(i),j}|E_{s(i)}) \prod_{i=k+1}^{N} p(x_{s(i),j}|\neg E_{s(i)})$, where $X_j = [x_{1j}, ..., x_{Nj},]$ represents the j's cross-sectional data for events $1, ..., N$.

Integrating out the hidden position k, one obtains: $p(X_j|S) = \sum_{k=0}^{N} p(k)p(X_j|S,k)$. And for all cross-sectional data, we have $p(X|S) = \prod_{j=1}^{J} p(X_j|S)$. Finally, Bayesian theorem is used to obtain posterior distribution for the ordering S: $p(S|X) = p(S)P(X|S)/P(X)$. Then MCMC sampling is used to sample from $p(S|X)$ to derive the most common ordering and the ordering variance (see Fig. 20).

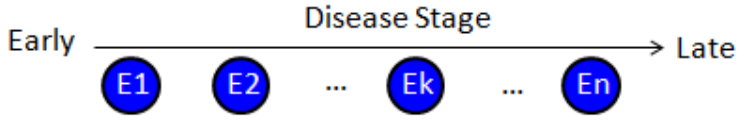


Figure 19: Disease progression as a series of events, each comprising a significant change in patient state.

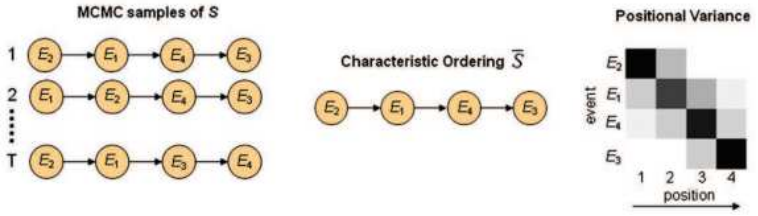


Figure 20: MCMC sampling for the event order S (from [18], used without permission).

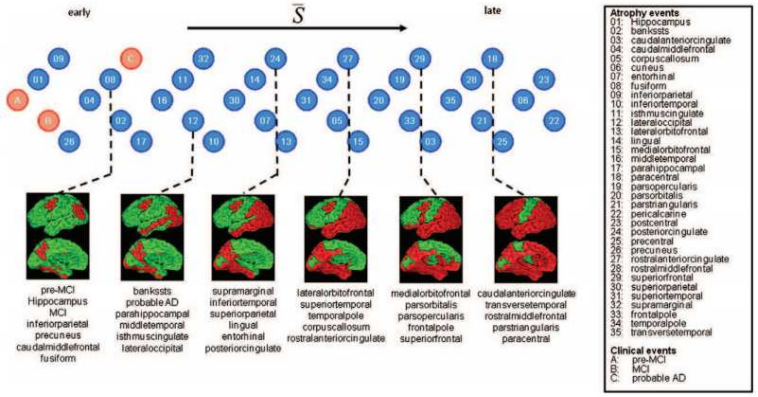


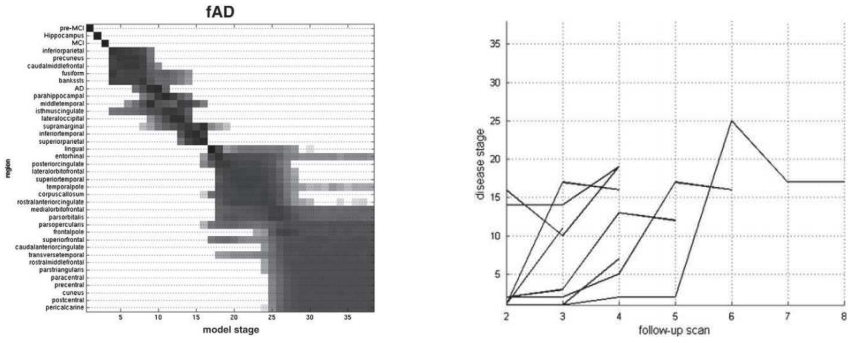
Figure 21: The found most common event ordering for Alzheimer's disease (from [18], used without permission).

The found most common event ordering is shown in Fig. 21 and the position variation derived from the MCMC samples is shown in Fig. 22(a). The learned event

model can be used to stage a cross-sectional data by computing the most probable current event location. The staging results for follow-up data for all patients are illustrated in Fig.22(b). The results provide new detail in the progression pattern of these diseases, while confirming known features, and give insight into the variability of progression over the patient cohort.

This event-based progression model has several merits but its current formulation has some limitations. First, it treats longitudinal data as cross-sectional data, and thus it does not use the full information from the longitudinal data in event positioning and measurement denoising. Second, the formulation ignores the duration information between visits, and thus no duration information between events can be statistically characterized.

We think that CT-HMM can also be utilized to deriving the most probable event ordering while overcoming the above two limitations. One idea is as follows. We can first using clustering techniques to find disease states from the cross-sectional data, and then model the temporal interactions among these states. As one can easily derive the probability of event occurrence in each state, the overall event ordering can be inferred from the learned model.



(a) (b)

Figure 22: (a)The distribution of positions for each event from MCMC samples. (b) The most probable event location for each follow-up visit for each patient. (From [18], used without permission).

2.4 Other multivariate longitudinal models in Biostatistics

Here, we will briefly discuss the univariate longitudinal models, and their extensions to multivariate longitudinal models developed in Biostatistics fields [5].

For univariate longitudinal analysis where there is one response variable and possibly many explanatory variables, fixed effect model, random effect model, and mixed effect model are well-established and widely used linear models in biostatistics. A fixed effect model represents the observed quantities in terms of explanatory variables that are treated as non-random. In contrast, random effects models and mixed effects models have some explanatory variables treated as if they arise from random causes [93]. In biostatistics, "fixed" and "random" effects are often used to refer to the population-average and subject-specific effects where the former is associated with fixed parameters while the latter is treated as random variables. In addition to the above three linear models, the generalized linear model (GLM) generalizes linear regression by allowing the linear model to be related to the response variable via a *link* function, such as exponential.

Multivariate longitudinal data means that there are repeated observations of multiple response variables. Special treatments are required to analyze this data as they are correlated over time and multiple responses measured at the same time are also correlated. [5] reviews three approaches for analyzing this kind of data. The first is to transform the multiple outcomes to a single summary variable and then use univariate longitudinal model on the summary outcome, which is conceptually the simplest, but the summary variable can be hard to interpret. The second method estimates regression coefficients without explicitly modeling the underlying covariance structure of the data. The third approach combines all the responses into a single joint multivariate model and is the most complicated.

Comparing the prior work to our MD CT-HMM, we find that many existing longitudinal methods in biostatistics for non-linear modeling require *synchronous* time

sequences, i.e., there are explicit or implicit correspondence between the timestamps of different patients. For example, tracking of a set of patients following the a surgical operation. For this time synchronous setting, *spline* model [5] can be useful for modeling nonlinear dynamics by putting a set of *knot points* on specified times. However, the longitudinal data for tracking chronic disease often have no time alignments among patients. This time synchronization is not required in our CT-HMM formulations, where the longitudinal data can start and begin at any state of the progression path. Furthermore, our MD CT-HMM explicitly models the state dependent transition behavior while conventional longitudinal model, such as mixed effect model, seeks a global mean path (average intercept and slope) and the variance of the heterogeneity (the random effect) which is much more compact but also have limited expressive power. In sum, our MD CT-HMM has benefits in modeling the underlying state-dependent evolution behavior and can naturally handle sequences without aligned reference times.

2.5 Conclusion

We reviewed several related state-based disease progression models and compare those models to our M-D CT-HMM in similar applications. CT-HMM models that can handle both continuous-time state transition process and noisy observation process are in principle better suited models for disease progression modeling with noisy clinical data than other continuous-time models without a separate observation process and other discrete-time models.

Our M-D CT-HMM which can analyze the temporal interactions of several disease markers/aspects with covariate modeling, is the extension of the conventional 1-D HMM-based disease progression models, which gives a more informed global view of the disease evolution. The novel parameter learning algorithms for CT-HMM

proposed in this dissertation also enable the applications of CT-HMM with large-state space, but without the restrictive assumptions on state transition timing, which forced in prior work. We also describe that CT-HMM can be utilized to model event transitions for Alzheimer's disease and both the event ordering and timing information can be inferenced, while the referred prior work ignores the time but only models the relative ordering.

A prior work using a hybrid CT-HMM for comorbidity-based disease staging modeling is discussed. This model does not directly use the key markers for defining disease progression, but is based on the onset patterns of hidden comorbidities inferenced from clinical observations. In contrast, our M-D CT-HMM defines the disease progression directly using the key markers while other clinical observations can be incorporated as covariates to construct individual-level transition rate matrix. We think that both models are valuable in defining disease progression from different perspective, which complement each other.

We also discuss a prior work that uses a double-layer CTMC for phase-type duration modeling in CTMC applications. This model may provide better duration modeling for CTMC models than the basic exponential distribution. However, further investigations are needed for assessing the use of this model in large-scaled state space and in CT-HMM settings.

We also briefly review several wide-used longitudinal models in Biostatistics. We find that our M-D CT-MM model has benefits of modeling non-linear progression from time-asynchronous records of patient data using composite disease states, which can flexibly capture multivariate interaction, while other non-linear models, such as spline, generally needs time alignments of longitudinal records.

Overall, our M-D CT-HMM introduces a novel way to analyze the temporal interactions of different disease evolution aspects (structural, functional, biochemical, etc.), and covariates can also be incorporated to learn and construct individualized

transition rates. Our MD CT-HMM is complementary to many existing disease models for providing comprehensive understanding of the global disease progression spectrum.

CHAPTER III

EFFICIENT CT-HMM PARAMETER LEARNING USING END-STATE CONDITIONED EXPECTATION EM

A *Continuous-Time* HMM (CT-HMM) is an HMM in which the transitions between hidden states and the observations can both occur at arbitrary continuous times [16]. It is thus more suitable for modeling continuously evolving process such as disease progression, and irregularly-arrived data such as clinical measurements than conventional discrete-time HMM in which both the state transitions and observational data occur at regular time intervals. The comparison of the underlying state transition process and observation process between discrete-time(DT) HMM and CT-HMM is illustrated in Fig. 23.

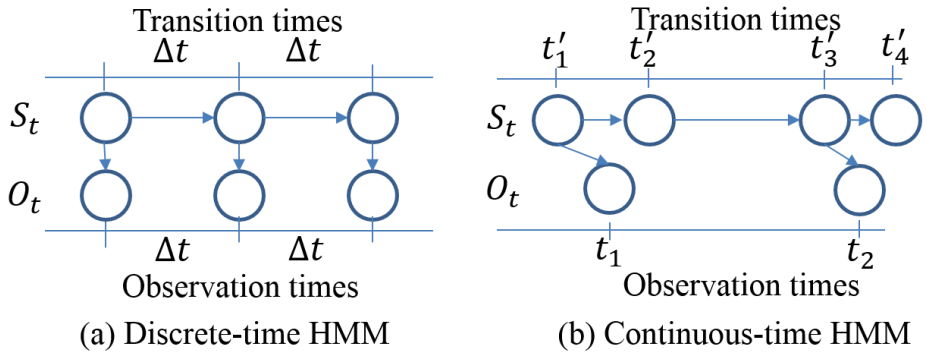


Figure 23: Comparison of discrete-time HMM and continuous-time HMM. (a) In conventional HMM, state transitions and observations happen at regular time intervals Δ_t. States are allowed to self-transition. (b) In continuous-time HMM, state transitions occur in unknown continuous-time $(t'_1, t'_2, ...)$ (always transition to other states), and observations also arrive at arbitrary times $(t_1, t_2, ...)$.

Unfortunately the additional modeling flexibility provided by the CT-HMM comes at the cost of a more complex inference procedure than DT-HMM. In CT-HMM, not only are the hidden states unobserved, but the *transition times* at which the hidden

states are changing are also unobserved. It is possible for multiple unobserved hidden state transitions to occur between two successive observations. An early parameter learning algorithm circumvent these challenges of explicit hidden path decoding by directly maximizing the data likelihood [29] using numerical optimization, but this method is limited to very small model structures. The lack of efficient learning algorithms has limited CT-HMM to applications of small models [29] or for large-scale models require unrealistic assumptions on state transition timing [42, 52]. There is a need for learning methods for CT-HMM which can scale to large state space [42].

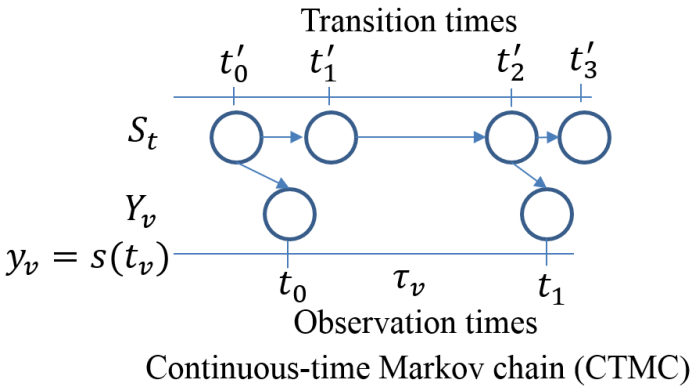


Figure 24: The graph shows a CTMC process under incomplete observations. Similar to CT-HMM, in a CTMC both the state transitions and observations occur at arbitrary continuous time. However, in a CTMC the observation data (y_v) are the hidden states at the observed moment ($y_v = s(t_v)$).

With the challenges of learning CT-HMM, one may resort to simply discretize the time horizon and just use discrete-time HMM. However, the time horizon for state changes in medical conditions can vary dramatically. In early stages of disease a state change might not occur for years, but in an acute phase they could occur very frequently. For states with very short expected dwelling times, the discrete time step needs to be sufficiently small. However, this might be inefficient for dealing with changes that occur once several years. On the other hand, if the discretization is too coarse, many transitions could be collapsed into a single one, obscuring the real dynamics in continuous-time. In contrast, CT model performs inference over

arbitrary timescales using a single matrix exponential. We believe CT is a better choice than DT for modeling continuous-time processes such as disease progression and clinical data.

To tackle the challenges of CT-HMM learning, if we consider a simpler case where the observation process observes the hidden states directly without noise but still at arbitrary times, then CT-HMM is equivalent to a continuous-time Markov chain (CTMC) model under incomplete observations (only knows the states at discrete observation times). An illustration of CTMC state transition process and incomplete observation process is shown in Fig. 24. EM algorithms for this CTMC setting to find maximum likelihood solution were developed recently [7, 64, 25]. They work by computing expected state durations and transition counts conditioned on each pair of successive state observations. In [64], efficient closed-form evaluations are obtained when the rate matrix can be diagonalized through an eigendecomposition (abbreviated as *Eigen*). Such an approach has been extended to CT-HMM [90]. However, the rate matrix is frequently not diagonalizable during learning, and this approach often fails in real data [64].

More recently, [25] demonstrated that the necessary conditional expectations can also be evaluated using two alternative approaches. One approach (abbreviated as *Expm*) uses a double-sized auxiliary matrix with respect to the rate matrix in computing the matrix exponential, while the other (abbreviated as *Unif*) utilizes the uniformization theory. For CTMC applications, it has been shown that *Expm* appears to be the most accurate, but is the most expensive [83]. Unfortunately, none of these methods have been explored or developed for CT-HMM, which has the ability to correctly account for measurement noise and offers more flexibility to clinicians in describing the relationship between measurements and disease states.

We present the first comprehensive framework for parameter learning in CT-HMM models of disease progression, which both extends and unifies prior work on CTMC

models. Based on the finite number of observations, our framework discretizes the estimation of posterior hidden state probabilities at observation times into a time-inhomogeneous HMM, and then incorporates two approaches to estimating the hidden state distributions. In the "hard" approach, the distribution over hidden states is approximated by the single most likely (Viterbi) decoding, which results in significant computational savings. The "soft" approach, in contrast, maintains the full distribution over the hidden states, gaining accuracy at the cost of increased computations. The benefits and drawbacks of these two approaches (*Soft* and *Hard*) combined with the three different computational approaches for evaluating the conditional expectations (*Expm, Unif, Eigen*) are analyzed and validated experimentally.

Our learning algorithms for CT-HMM are evaluated on both synthetic data and real-world disease datasets. For the simulation, we assess the learning accuracy of our methods with respect to different observation intervals, observation counts, and varying noise levels. We find that soft methods are more robust than hard methods especially when the noise levels are high. The running time of our EM algorithms are also compared under different synthetic or real data. We find that *Soft(Expm)* method is the most time efficient under soft EM, while *Hard(Unif)* method can be the most economic in hard EM due to its decomposibility in evaluating only the required end-states and better generality than Eigen method.

For the two real datasets from glaucoma and Alzheimer's disease, we demonstrate applications include visualizations of the progression model and predictions of future progression. Our prediction results outperform the state-of-the-art method [62] for glaucoma prediction, which shows the practical value of CT-HMM for longitudinal disease prognosis and management.

3.1 Basic formulations of CTMC and CT-HMM

A continuous-time Markov chain (CTMC) is defined by a finite and discrete state space S, a state transition rate matrix Q, and an initial state probability distribution π. The elements q_{ij} in Q are non-negative and describe the rate (instantaneous risk) the process transitions from state i to j, for $i \neq j$ and q_{ii} are specified such that each row of Q sums to zero (the holding time parameter $q_i = \sum_{j \neq i} q_{ij}$ and $q_{ii} = -q_i$) [16]. In the most general form, Q can be time-inhomogeneous, and one can express q_{ij} as:

$$q_{ij}(t, z(t)) = \lim_{\delta t \approx 0} \frac{p(s(t + \delta t) = j | s(t) = i)}{\delta t}$$

where $s(t)$ represents the state at time t, and $z(t)$ represents a set of individual-level constant or time-varying covariates (explanatory variables), such as age, and treatment history. In a time-homogeneous process, in which the q_{ij} are independent of t, the sojourn time (a single period of occupancy) in each state i is exponentially-distributed with the holding time parameter q_i as $f(t) = q_i e^{-q_i t}$, which has mean $1/q_i$. The probability that an individual in state i moves next to state j is q_{ij}/q_i, for $i \neq j$. The *Markov* assumption states that the future of the process depends only on the current state and not on the history. For a thorough review of continuous-time Markov chain (CTMC) theory, please reference Cox and Miller [16].

The continuous-time Markov model for panel-observed data (observations arrived at arbitrary times) was first described by Kalbfleisch and Lawless [32] and Kay [33]. In the most general form, the transition probability $p_{ij}(t_0, t)$ represents that given the state at time instant t_0 is i, the probability of being in state j at time instant t. The transition probability matrix $P(t_0, t)$ is calculated in terms of $Q(t)$ using the *Kolmogorov differential equations* [16] as follows:

$$\frac{dP(t_0, t)}{dt} = P(t_0, t)Q(t).$$

The general solution to the Kolmogorov differential equations is:

$$P(t_0, t) = \sum_{k=0}^{\infty} \int_{t_1=t_0}^{t} \int_{t_2=t_1}^{t} \cdots \int_{t_k=t_{k-1}}^{t} Q(t_1)Q(t_2)\cdots Q(t_k)dt_1 dt_2 \cdots d_{t_k}$$

where k represents the number of possible state jumps which ranges from 0 to ∞, and $t_1 \cdots t_k$ represents the times of these jumps.

If the transition intensity matrix Q is constant over the time interval (t_0, t), as in a time-homogeneous process, then $P(t_0, t)$ can be expressed as $P(\tau)$ where $\tau = t - t_0$ represent the time interval. In this case, the Kolmogorov equations are solved by the matrix exponential of Q scaled by the time interval τ:

$$P(\tau) = e^{\tau Q}.$$

The matrix exponential is defined by the same power series $e^X = 1 + X^2/2! + X^3/3! + ...$ as the scalar exponential, except that each term X^k is defined by matrix product, not element-wise scalar product. The matrix exponential can be solved by *Pade* approximation [23].

We now describe CT-HMM. In contrast to CTMC, where the states are directly observed, in CT-HMM, none of the state is directly observed, but observational data o are generated conditioned on the hidden states s from a data emission model $p(o|s)$ [16] [29]. This emission model is used to capture the fact that the real world data is a noisy or transformed version of the true state. Furthermore, the observations $(o_0, o_1, \ldots, o_V)$ are typically collected in only at discrete and irregular time points $(t_0, t_1, \ldots, t_V)$. In other words, there are two levels of hidden information in CT-HMM. First, at the observation times, the true state of the Markov chain is not observed. Second, the Markov chain between two consecutive observations is hidden, and the Markov chain may have visited other states before it reaches the state at the observation time. While CT-HMM is in principle a better model for real world data, the two levels of complication makes the parameters in CT-HMM even more challenging to estimate.

3.2 Prior work: Maximum Likelihood Estimation in CTMC

When a realization of the CTMC is *fully* observed, meaning that one can observe every state transition time, $(t'_0, t'_1, \ldots, t'_{V'})$, and the corresponding state $Y = \{y_0 = s(t'_0), ..., y_{V'} = s(t'_{V'})\}$, where $s(t)$ denotes the state at time t, the complete likelihood of the data is

$$CL = \prod_{v'=0}^{V'-1} (q_{y_{v'}, y_{v'+1}} / q_{y_{v'}})(q_{y_{v'}} e^{-q_{y_{v'}} \tau_{v'}}) = \prod_{v'=0}^{V'-1} q_{y_{v'}, y_{v'+1}} e^{-q_{y_{v'}} \tau_{v'}} \tag{1}$$

$$= \prod_{i=1}^{|S|} \prod_{j=1, j \neq i}^{|S|} q_{ij}^{n_{ij}} e^{-q_i \tau_i} \tag{2}$$

where $\tau_{v'} = t'_{v'+1} - t'_{v'}$ is the time interval between the two successive observations, n_{ij} is the total number of transitions from state i to j, and τ_i is the total amount of time the chain remains in state i. In this setting, the optimal parameters have a simple analytical solution:

$$q_{ij} = \frac{n_{ij}}{\tau_i}, i \neq j \quad \text{and} \quad q_{ii} = -\sum_{j \neq i} q_{ij}. \tag{3}$$

However, a realization of the CTMC is typically observed only at discrete and irregular time points $(t_0, t_1, ..., t_V)$ with the corresponding state sequence $Y = \{y_0 = s(t_0), ..., y_V = s(t_V)\}$. The process between the two consecutive observations is *hidden*, and the Markov chain may have visited other states before it reaches the observed state. Thus both the transition counts between any two states i and j, denoted as n_{ij}, and the total duration in a state i, denoted as τ_i, are unknown. In Fig. 25 the comparison of the state transition and observation process between fully and partially-observed CTMCs is illustrated.

In order to express the likelihood of these incomplete observations, one needs to use the state transition probability matrix, $P(t) = e^{Qt}$, where $P_{ij}(t)$, the entry (i, j) in $P(t)$, is the probability that the process is in state j after time t given that it is in state i at time 0. This quantity takes into account all possible intermediate states and state durations visited between i and j but not observed. Then the likelihood of

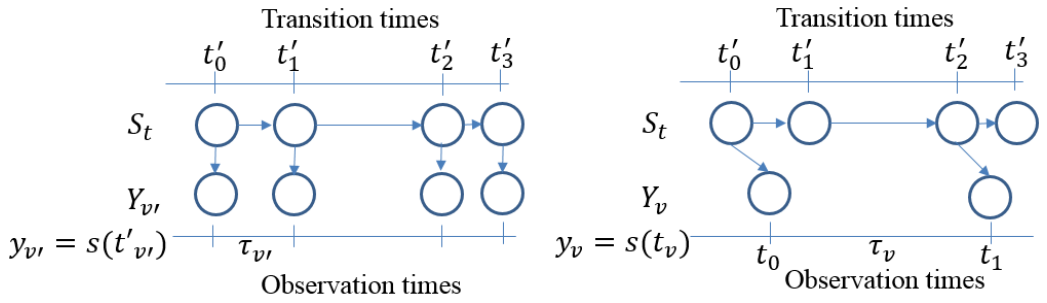


Figure 25: Comparison of the state transition and observation process between (a) fully-observed and (b) partially-observed CTMCs is illustrated.

the data is

$$L = \prod_{v=0}^{V-1} P_{y_v, y_{v+1}}(\tau_v) = \prod_{v=0}^{V-1} \prod_{i,j=1}^{|S|} P_{ij}(\tau_v)^{\mathbb{I}(y_v=i, y_{v+1}=j)} = \prod_{\Delta=1}^{r} \prod_{i,j=1}^{|S|} P_{ij}(\tau_\Delta)^{C(\tau=\tau_\Delta, y_v=i, y_{v+1}=j)} \quad (4)$$

where τ_Δ, $\Delta = 1, ..., r$, represents r unique values among all time intervals τ_v from the data, $\mathbb{I}(y_v = i, y_{v+1} = j)$ is an indicator function that is 1 if the condition is true, otherwise it is 0, and $C(\tau = \tau_\Delta, y_v = i, y_{v+1} = j)$ is the total counts from every two successive visits when the condition is true. Note that there is no known analytical maximizer of L with respect to Q.

An EM algorithm is designed recently to tackle this challenge [64], where the expected log-complete likelihood in the E-step takes the form:

$$\sum_{i=1}^{|S|} \sum_{j=1, j\neq i}^{|S|} \{\log(q_{ij}) E[n_{ij}|Y, \hat{Q}_0] - q_i E[\tau_i|Y, \hat{Q}_0]\},$$

where $\hat{Q}_0$ is the current estimate for Q, and $E[n_{ij}|Y, \hat{Q}_0]$ and $E[\tau_i|Y, \hat{Q}_0]$ are the expected state transition count and duration given the observation Y and the current transition rate matrix $\hat{Q}_0$, respectively. This expected log-likelihood is derived from the complete data likelihood (CL) for algebraical simplicity for derivation without loss of generality [7][64].

Once the two kinds of statistics $E[n_{ij}|Y, \hat{Q}_0]$ and $E[\tau_i|Y, \hat{Q}_0]$ are computed, the parameter $\hat{Q} = (q_{ij})$ in the M-step can be updated as

$$\hat{q}_{ij} = \frac{E[n_{ij}|Y, \hat{Q}_0]}{E[\tau_i|Y, \hat{Q}_0]}, i \neq j \quad \text{and} \quad \hat{q}_{ii} = -\sum_{j\neq i} \hat{q}_{ij}. \quad (5)$$

Now the main computational challenge is to evaluate $E[n_{ij}|Y, \hat{Q}_0]$ and $E[\tau_i|Y, \hat{Q}_0]$. By exploiting the properties of the *Markov* process, the expectations can be decomposed as [7]:

$$E[n_{ij}|Y, \hat{Q}_0] = \sum_{v=0}^{V-1} E[n_{ij}|y_v, y_{v+1}, \hat{Q}_0] \tag{6}$$

$$= \sum_{v=0}^{V-1} \sum_{k,l=1}^{|S|} \mathbb{I}(y_v = k, y_{v+1} = l) E[n_{ij}|y_v = k, y_{v+1} = l, \hat{Q}_0] \tag{7}$$

$$= \prod_{\Delta=1}^{r} \prod_{k,l=1}^{|S|} C(\tau = \tau_\Delta, y_v = k, y_{v+1} = l) E[n_{ij}|y_v = k, y_{v+1} = l, \tau_v = \tau_\Delta, \hat{Q}_0] \tag{8}$$

where $\mathbb{I}(y_v = k, y_{v+1} = l) = 1$ if $s(t_v) = k$ and $s(t_{v+1}) = l$, otherwise it is 0, and $C(\tau = \tau_\Delta, y_v = k, y_{v+1} = l)$ is the total counts from every two successive visits when the condition is true. The derivation for $E[\tau_i|Y, \hat{Q}_0]$ can be done similarly. Thus, the computation now boils down to computing the end-state conditioned expectations $E[n_{ij}|y_v = k, y_{v+1} = l, \hat{Q}_0]$ and $E[\tau_i|y_v = k, y_{v+1} = l, \hat{Q}_0]$, for all required $k, l, i, j \in S$.

A key result enabling efficient computation of end-state conditioned statistics of CTMC is the derivation of their integral form in recent literature [24]:

$$E[n_{ij}|s(0) = k, s(t) = l, Q] = \frac{q_{i,j}}{(e^{Qt})_{k,l}} \int_0^t (e^{Qx})_{k,i} (e^{Q(t-x)})_{j,l} \, dx = \frac{q_{i,j} \tau_{k,l}^{i,j}(t)}{(e^{Qt})_{k,l}} \tag{9}$$

$$E[\tau_i|s(0) = k, s(t) = l, Q] = \frac{1}{(e^{Qt})_{k,l}} \int_0^t (e^{Qx})_{k,i} (e^{Q(t-x)})_{i,l} \, dx = \frac{\tau_{k,l}^{i,i}(t)}{(e^{Qt})_{k,l}} \tag{10}$$

where $\tau_{k,l}^{i,j}(t) = \int_0^t (e^{Qx})_{k,i} (e^{Q(t-x)})_{j,l} \, dx$. It is then important to compute $\tau_{k,l}^{i,i}(t)$ and $\tau_{k,l}^{i,j}(t)$ efficiently.

[64] observed that the calculation of $\tau_{k,l}^{i,j}(t)$ can be done in closed-form if Q is diagonalizable and one can act eigendecomposition on Q (denoted hereafter as the *Eigen* method) [24, 64, 65, 25]. Consider the eigendecomposition of $Q = UDU^{-1}$, where the matrix U consists of all eigenvectors to the corresponding eigenvalues of Q in the diagonal matrix $D = diag(\lambda_1, ..., \lambda_n)$. Then we have $e^{Qt} = Ue^{Dt}U^{-1}$ and the integral can be written as: $\tau_{k,l}^{i,j}(t) = \sum_{p=1}^{n} U_{kp} U_{pi}^{-1} \sum_{q=1}^{n} U_{jq} U_{ql}^{-1} \Psi_{pq}(t)$,where the symmetric matrix $\Psi(t) = [\Psi_{pq}(t)]_{p,q \in S}$ is defined as: $\Psi_{pq}(t) = te^{t\lambda_p}$ if $\lambda_p = \lambda_q$, and $\Psi_{pq}(t) = \frac{e^{t\lambda_p} - e^{t\lambda_q}}{\lambda_p - \lambda_q}$ if $\lambda_p \neq \lambda_q$. However, not all Q matrices during learning are

diagonalizable. We will then discuss alternatives and more general methods that recently discovered to compute $\tau_{k,l}^{i,j}(t)$ in Section **??**.

3.2.1 Evaluation of end-state conditioned expected statistics

Besides *Eigen* method, we now describe two alternative methods, one called matrix exponential, and the other called uniformization method from [25], to evaluate $\tau_{k,l}^{i,j}(t) = \int_0^t (e^{Qx})_{k,i}(e^{Q(t-x)})_{j,l}dx$, in order to compute the sufficient statistics

$$E[\tau_i|s(0)=k, s(t)=l] = \frac{\tau_{k,l}^{i,i}(t)}{P_{k,l}(t)}, \quad E[n_{i,j}|s(0)=k, s(t)=l] = q_{ij}\frac{\tau_{k,l}^{i,j}(t)}{P_{k,l}(t)}.$$

to update Q parameters during learning.

Matrix Exponential for evaluating the integral (abbreviated as *Expm*):
It is first noted in [25] recently that the calculation of the integral $\tau_{k,l}^{i,j}(t) = \int_0^t (e^{Qx})_{k,i}(e^{Q(t-x)})_{j,l}dx$ can be done using the theory developed in Van Loan [89]. Consider the problem of evaluating the integral of the form $\int_0^t e^{Qx}Be^{Q(t-x)}dx$, where B is a matrix of the same dimension as Q, [89] has shown that the integral result is the upper-right corner of e^{At} as:

$$\int_0^t e^{Qx}Be^{Q(t-x)}dt = (e^{At})_{(1:n),(n+1):(2n)}$$

where $A = \begin{bmatrix} Q & B \\ 0 & Q \end{bmatrix}$, and n is the dimension of matrix Q.

Now, the key observation from [25] is by setting up $B = U(i,j)$, where U is the matrix with a 1 in the (i,j)th entry and 0 elsewhere, and then the integral results in a matrix which has $\tau_{k,l}^{i,j}(t)$ for all k, l in the corresponding matrix entry. That is,

$$\int_0^t e^{Qx}Be^{Q(t-x)}dx = \int_0^t e^{Qx}U(i,j)e^{Q(t-x)}dx \tag{11}$$

$$= \begin{bmatrix} \tau_{1,1}^{i,j}(t) & \tau_{1,2}^{i,j}(t) & \cdots & \tau_{1,|S|}^{i,j}(t) \\ \tau_{2,1}^{i,j}(t) & \tau_{2,2}^{i,j}(t) & \cdots & \tau_{2,|S|}^{i,j}(t) \\ \cdots & \cdots & \cdots & \cdots \\ \tau_{|S|,1}^{i,j}(t) & \tau_{1,2}^{i,j}(t) & \cdots & \tau_{|S|,|S|}^{i,j}(t) \end{bmatrix} = (e^{At})_{(1:n),(n+1):(2n)} \tag{12}$$

where $A = \begin{bmatrix} Q & U(i,j) \\ 0 & Q \end{bmatrix}$. So one can do matrix exponential on A and retrieve the resulting top-right corner to get the desired $\tau_{k,l}^{i,j}(t)$ values.

Uniformization method (abbreviated as *Unif*) [25]:

The uniformization theory shows how CTMC and DTMC is equivalence subordinated to a *Poisson* process (see [70]) and gives an alternative description of the CTMC process. It was used as an alternative method for computing matrix exponential $P(t) = e^{Qt}$ [31] as follows. Define $\hat{q}_i = \max_i q_i$, and matrix $R = \frac{Q}{\hat{q}_i} + I$, where I is the identify matrix. Then, we have

$$e^{Qt} = e^{\hat{q}_i(R-I)t} = \sum_{m=0}^{\infty} R^m \frac{(\hat{q}_i t)^m}{m!} e^{-\hat{q}_i t} = \sum_{m=0}^{\infty} R^m Pois(m; \hat{q}_i t) \tag{13}$$

, where $Pois(m; \hat{q}_i t)$ is the probability of m occurrences from a Poisson distribution with mean $\hat{q}_i t$. In practice, a truncation point of the infinite sum has to determined. One benefits of using this method is that when one has same Q, but varying t to be evaluated, the R power series can be precomputed and reused, and thus can be very efficient in this scenario.

Then the necessary expectations can be expressed by directly inserting the e^{Qt} definition into the integral [25]:

$$E[n_{ij}, s(t) = l | s(0) = k] = R_{ij} \sum_{m=1}^{\infty} [\sum_{n=1}^{m} (R^{n-1})_{ki} (R^{m-n})_{jl}] Pois(m; \hat{q}t) \tag{14}$$

$$E[\tau_i, s(t) = l | s(0) = k] = \sum_{m=0}^{\infty} \frac{t}{m+1} [\sum_{n=0}^{m} (R^{n})_{ki} (R^{m-n})_{il}] Pois(m; \hat{q}t) \tag{15}$$

The main difficulty in using *Unif* in practice is to determine a truncation point of the infinite sum. For large values of $\hat{q}t$, we have $Pois(\hat{q}t) \approx N(\hat{q}t, \hat{q}t)$, where $N(\mu, \sigma^2)$ is the normal distribution and one can then bound the truncation error from the tail of Poisson by using cumulative normal distribution [83]. A truncation point

at $M = \ulcorner 4 + 6\sqrt{\hat{q}t} + (\hat{q}t) \urcorner$ is suggested in [83] to have error bound of 10^{-8} when approximate e^{Qt}, which we will adopt in our unif-based learning algorithm.

3.3 Prior work: parameter learning for CT-HMM

The commonly-used existing methods to estimate parameters for CT-HMM are reviewed below. For Maximum Likelihood Estimation (MLE) based methods, numerical optimization, such as the BroydenFletcherGoldfarbShanno (BFGS) algorithm, is generally adopted to directly optimize the data likelihood function L below [29] (used in in the **msm R** package):

$$L = \sum_{Y=\{s(t_0),\ldots,s(t_V)\}} \prod_{v=1}^{V} [P(\tau_v)]_{s(t_{v-1}),s(t_v)} \prod_{v=0}^{V} p(o_v|s(t_v)) \tag{16}$$

which considers all possible hidden state sequences Y at observation times, and $P(\tau_v) := e^{\hat{Q}\tau_v}, \tau_v = t_{v+1} - t_v$ (For simplicity, likelihood assumes one longitudinal data is shown.The data likelihood for all individuals is computed by multiplying the individual likelihood together). L can be evaluated using Forward-Backward algorithm as in discrete-time HMM case.

Recently, the EM algorithm developed for CTMC with eigen-decomposition method to evaluate the end-state conditioned expectations (as discussed in Sec. 3.2) has been applied to learn the rate matrix in a hybrid CT-HMM system [90]. However, the eigen method can often fail during learning and is not a general method. Another EM method to find MLE developed in [42] assumes that the state transition can only occurs exactly at the observation times, and thus embeds a discrete-time Markov chain into the continuous-time setting. However, this assumption is too restrictive to capture the real system dynamics and can introduce artifacts. There is also Bayesian method with prior using Markov Chain Monte Carlo sampling [14] to estimate the probability distribution of parameters.

In our experiences these existing methods (direct numerical optimization and MCMC sampling) which do not exploit much of the problem properties, do not scale

up well with the state space (the time complexity will be discussed below). There is a need for efficient and genearl parameter learning method for large-scale CT-HMM without restrictive assumptions on state transitions.

3.4 Maximum Likelihood Estimation for CT-HMM

In this section, we describe our framework to do maximum likelihood estimation for CT-HMM, by extending and unifying the work from CTMC. In contrast to CTMC, where the states are directly observed, in CT-HMM, none of the state is directly observed, but observational data o are generated conditioned on the hidden states s from a data emission model $p(o|s)$. This emission model is used to capture the fact the real world data is a noisy or transformed version of the true state. Furthermore, the observations $(o_0, o_1, \ldots, o_V)$ are typically collected in only at discrete and irregular time points $(t_0, t_1, \ldots, t_V)$.

There are two levels of hidden information in CT-HMM. First, at the observation times, the true state of the Markov chain are also not observed. Second, the Markov chain between two consecutive observations is also hidden, which can have zero or many state jumps. While CT-HMM is a better model for real world data, the two levels of complication makes the parameters in CT-HMM even more challenging to estimate. We are not aware of any previous work along this line and we will design a new EM algorithm to address this challenge.

3.4.1 Challenges for learning CT-HMM

A fully observed CT-HMM contains four sequences of information: the underlying transition time $(t'_0, t'_1, \ldots, t'_{V'})$, the corresponding state $Y = \{y_0 = s(t'_0), ..., y_{V'} = s(t'_{V'})\}$ of the hidden Markov chain, and the observed data $O = (o_0, o_1, \ldots, o_V)$ at time $T = (t_0, t_1, \ldots, t_V)$. Their joint likelihood can be written as

$$CL = \prod_{v'=0}^{V'-1} q_{y_{v'}, y_{v'+1}} e^{-q_{y_{v'}}\tau_{v'}} \prod_{v=0}^{V} p(o_v|s(t_v)) = \prod_{i=1}^{|S|} \prod_{j=1, j\neq i}^{|S|} q_{ij}^{n_{ij}} e^{-q_i\tau_i} \prod_{v=0}^{V} p(o_v|s(t_v)). \quad (17)$$

(In this thesis, we will focus on the estimation of the transition rate matrix Q of the Markov chain. Those parameters in the emission model $p(o|s)$ can be estimated in a similar way as discrete time HMMs, as in Eqn (40c), (52-54) of [68].)

In the EM algorithm, given a current estimate of the parameter $\hat{Q}_0$, the expected complete log-likelihood in E-step takes the form:

$$\sum_{i=1}^{|S|}\sum_{j=1,j\neq i}^{|S|}\{\log(q_{ij})E[n_{ij}|O,T,\hat{Q}_0]-q_iE[\tau_i|O,T,\hat{Q}_0]\}+\sum_{v=0}^{V}E[\log p(o_v|s(t_v))|O,T,\hat{Q}_0].$$

In the M-step, taking derivative with respect to q_{ij}, and we have

$$\hat{q}_{ij}=\frac{E[n_{ij}|O,T,\hat{Q}_0]}{E[\tau_i|O,T,\hat{Q}_0]}, i\neq j \quad \text{and} \quad \hat{q}_{ii}=-\sum_{j\neq i}\hat{q}_{ij}, \tag{18}$$

where we need to compute the even more challenging quantity of $E[n_{ij}|O,T,\hat{Q}_0]$ and $E[\tau_i|O,T,\hat{Q}_0]$. We show that they can be computed as:

$$E[n_{ij}|O,T,\hat{Q}_0]=\sum_{v=1}^{V-1}\sum_{k,l=1}^{|S|}p(s(t_v)=k,s(t_{v+1})=l|O,T)E[n_{ij}|s(t_v)=k,s(t_{v+1})=l,\hat{Q}_0] \tag{19}$$

$$E[\tau_i|O,T,\hat{Q}_0]=\sum_{v=1}^{n-1}\sum_{k,l=1}^{|S|}p(s(t_v)=k,s(t_{v+1})=l|O,T)E[\tau_i|s(t_v)=k,s(t_{v+1})=l,\hat{Q}_0] \tag{20}$$

where $p(s(t_v)=k,s(t_{v+1})=l|O,T)$ is the probability of being in state k at visit v and in state l at visit $(v+1)$ given data O and T. Now the key computation of the EM algorithm boils down to computing the posterior expectations $p(s(t_v)=k,s(t_{v+1})=l|O,T)$ and $E[n_{ij}|s(t_v)=k,s(t_{v+1})=l,\hat{Q}_0]$ and $E[\tau_i|s(t_v)=k,s(t_{v+1})=l,\hat{Q}_0]$ given $\hat{Q}_0$. All these quantities are challenging to compute. Computing the posterior state probabilities $p(s(t_v)=k,s(t_{v+1})=l|O,T)$ requires considering all possible state transition sequences from k to l, as well as the variable time intervals between each transition. Similar high-dimensional integrals are also needed for computing the end-state conditioned statistics, $E[n_{ij}|s(t_v)=k,s(t_{v+1})=l,\hat{Q}_0]$ and $E[\tau_i|s(t_v)=k,s(t_{v+1})=l,\hat{Q}_0]$. We now present our solution to these challenges.

Posterior State Probability at Observation Time $p(s(t_v)=k,s(t_{v+1})=l|O,T)$**:**

During the computation of this quantity, it is important to avoid explicit enumeration of all possible state assignments and variable time intervals. In order to efficiently

tackle this problem, we note that the posterior state probabilities are needed only for the times where we observed the data. We will exploit this key characteristic to tackle the problem by constructing a discrete *time-inhomogeneous* hidden Markov model.

More specifically, given current estimate $\hat{Q}_0$, O and T, we will divide the time into V intervals, each with duration $\tau_v = t_v - t_{v-1}$. Then make use of the Markov property, we will associate each interval with a state transition matrix $P(\tau_v) := e^{\hat{Q}_0 \tau_v}$. Together with the emission model $p(o|s)$, then we have a discrete time-inhomogeneous hidden Markov model with join likelihood of the $V+1$ hidden and observation as

$$\sum_{Y=\{s(t_0),\ldots,s(t_V)\}} \prod_{v=1}^{V} [P(\tau_v)]_{s(t_{v-1}),s(t_v)} \prod_{v=0}^{V} p(o_v|s(t_v)). \tag{21}$$

The above discrete-time Markov chain allows us to reduce the computation of $p(s(t_v) = k, s(t_{v+1}) = l|O,T)$ to familiar operations. The *forward-backward* algorithm can be used to compute the posterior probability of the hidden states [68] (referred as soft approach). The Viterbi algorithm can also be used to obtain a maximum-a-posteriori (MAP) assignment of hidden states using the state-optimized likelihood as below,

$$max_{Y=\{s(t_0),\ldots,s(t_V)\}} \prod_{v=1}^{V} [P(\tau_v)]_{s(t_{v-1}),s(t_v)} \prod_{v=0}^{V} p(o_v|s(t_v)) \tag{22}$$

and approximate the desired quantity by only using the MAP state sequence referred as hard approach).

Note that our discretization here calculates exact statistics and is efficient. The time inhomogeneous transition matrix $P(tau_v) = exp(Qtau_v)$ between two visits summarizes all possible hidden transition paths and times between every end-state pair given a time interval, and the dynamic programming efficiently summarizes probabilities of all possible hidden states at observation times.

End-State Conditioned Statistics of CTMC: In Section 3.2.1, we have explained three methods to compute the end-state conditioned statistics $E[n_{ij}|s(t_v) =$

$k, s(t_{v+1}) = l, \hat{Q}_0]$ and $E[\tau_i | s(t_v) = k, s(t_{v+1}) = l, \hat{Q}_0]$, which are *Eigen, Expm, Unif* methods. Eigen method can only be stable when Q matrix is diagnalizable, which may often not be the case during EM learning. The two other recently-discovered alternatives [25], one is based on performing matrix exponential (*Expm* method) on a double-sized auxiliary matrix [89], and another is called *uniformization* method, which origins from the uniformization theory to connect CTMC and DTMC, both have not been explored or studies for CT-HMM before.

In the next Section, we will propose novel EM learning algorithms considering different combinations of the above methods and explore the best choices under different criteria and data properties.

3.5 EM algorithms for CT-HMM

The main EM algorithm is shown in Algorithm 1. Depending on the combination of the hard/soft assignment during the discrete time-inhomogeneous HMM stage, and the three different approaches of computing the end-state conditioned statistics, we will obtain many different variants of the EM algorithm for CT-HMM. We will use the term *Soft* and *Hard* to represent that it is *soft/hard* decoding of the hidden states for the actual visits.

Algorithm 1 CT-HMM Parameter learning (Soft/Hard)

1: **Input:** data $O = (o_0, ..., o_V)$ and $T = (t_0, \ldots, t_V)$, state set S, edge set L
2: **Output:** transition rate matrix $Q = (q_{ij})$
3: Find all distinct time intervals t_Δ, $\Delta = 1, ..., r$, from T
4: Compute $P(t_\Delta) = e^{Qt_\Delta}$ for $t_\Delta \Rightarrow O(rS^3)$
5: **repeat**
6: Compute $p(v, i, j) = p(s(t_v) = i, s(t_{v+1}) = j | O, T) = p(v, i, j)$, for $v = 0, ..., V-1$, and the complete/state-optimized data likelihood l by using *Forward-Backward*(soft)/ *Viterbi*(hard) $\Rightarrow O(VS^2)$
7: Create soft count table $C(\Delta, i, j)$ from $p(v, i, j)$ by summing prob. from visits of same t_Δ
8: Use Expm, Unif or Eigen algorithm here
9: Update $q_{ij} = \frac{E[n_{ij}|O,T]}{E[\tau_i|O,T]}$, and $q_{ii} = -\sum_{i \neq j} q_{ij}$
10: **until** likelihood converges

Once the MLE solution has been found, the approximates standard errors of the parameters can be derived from the inverse of the matrix of second derivatives (Hessian) of the log-likelihood function at the maximized solution, which can be computed by using finite differences [26].

3.5.1 Algorithm: Soft(Expm), Hard(Expm) EM

Algorithm 2 shows the algorithm that uses the Expm method for compute end-state conditioned statistics. In Algorithm 1 line 7, we group probabilities from successive visits of same time interval and same specified end-states in order to save computation times. In Algorithm 2, we put the loop for different time intervals (line 2, 8) within the loop for each state and edge (line 1, 7) on purpose. As the A matrix does not change with time t_Δ, if use *scaling and squaring* method [23] for *expm* operator (used in Matlab), one can easily modify the $expm(A)$ function to be $expm(A,t) = e^{tA}$ and caches the intermediate results of A^2, A^4, and A^6 to be reused for computing e^{tA} for a different t. The overall time complexity per iteration is in line 13, where r is the number of distinct time interval, S is the number of states, and L is the number of edges.

Algorithm 2 Expm Algorithm

1: **for** each state i in S **do**
2: **for** $\Delta = 1$ **to** r **do**
3: $D_i = \frac{[e^{t_\Delta A}]_{(1:n),(n+1):(2n)}}{P_{kl}(t_\Delta)}$, where $A = \begin{bmatrix} Q & I(i,i) \\ 0 & Q \end{bmatrix}$ $\Rightarrow O((2S)^3)$
4: $E[\tau_i|O,T] \mathrel{+}= \sum_{(k,l)\in L} C(\Delta,k,l) D_i(k,l)$
5: **end for**
6: **end for**
7: **for** each link (i,j) in L **do**
8: **for** $\Delta = 1$ **to** r **do**
9: $N_{ij} = \frac{q_{ij}[e^{t_\Delta A}]_{(1:n),(n+1):(2n)}}{P_{kl}(t_\Delta)}$, where $A = \begin{bmatrix} Q & I(i,j) \\ 0 & Q \end{bmatrix}$ $\Rightarrow O((2S)^3)$
10: $E[n_{ij}|O,T] \mathrel{+}= \sum_{(k,l)\in L} C(\Delta,k,l) N_{ij}(k,l)$
11: **end for**
12: **end for**
13: Soft/Hard: $O(r(2S)^4 + rL(2S)^3)$

3.5.2 Algorithm: Soft(Unif), Hard(Unif) EM

In Algorithm 3, we used *Unif* method for compute end-state conditioned statistic for CTMC in Algorithm 3. In line 6 and 10, $S_{k\to l}$ (and $L_{k\to l}$) represents the intermediate states (edges) that can be passed from state k to l. The state accessibility table can be precomputed using *Dijkastra*'s shortest path algorithm in $O(S^2)$. The main benefits of *Unif* in evaluating all expectations is that the R power series $(R, R^2, ..., R^{\hat{M}})$, can be precomputed (line 2) and reused, so that no additional matrix multiplications is needed.

One main property of *Unif* in evaluating expectation is its $O(M^2)$ complexity, which is not related to state space size S. It can also evaluate the expectations for just the two specified end-states. In hard EM the soft count table $C(\Delta, k, l)$ (in line 5) becomes sparse ($\leq min(V, rS^2)$ entries have positive values), and thus *Unif* in hard EM becomes more time efficient than soft EM. One possible downside of *Unif* is that if $\hat{q}_i t$ is very large, so is the truncation point M, then the computation can become very time consuming. Thus, we find that *Unif*'s running time performance highly depends on the data and the underlying Q values. The time complexity analysis is detailed in Algorithm 3 line 16.

3.5.3 Algorithm: Soft(Eigen), Hard(Eigen) EM

In Algorithm 4, the Eigen method is presented. *Eigen* method also has the flexibility in evaluating the expectations only for the specified end-states, and thus it can be more efficient in hard than in soft EM. The main problem of *Eigen* is that it is not a stand-alone algorithm. When Q is not diagonalizable in any iteration, one needs alternative methods for that run.

3.5.4 Comparison of time complexity of all methods

The time complexity comparison of all methods is listed in Table 3. When comparing Soft EM methods, we find that *S(Expm)* has better time complexity than *S(Eigen)*

Algorithm 3 Unif Algorithm

Set $\hat{t} = max\ \ t_\Delta$; set $\hat{q} = max_i q_i$.
Let $R = Q/\hat{q} + I$. Compute $R, R^2, ..., R^{\hat{M}}$, $\hat{M} = \ulcorner 4 + 6\sqrt{\hat{q}\hat{t}} + (\hat{q}\hat{t})\urcorner \Rightarrow O(\hat{M}S^3)$
for $\Delta = 1$ **to** r **do**
  $M = \ulcorner 4 + 6\sqrt{\hat{q}t_\Delta} + (\hat{q}t_\Delta)\urcorner$; set $t = t_\Delta$
  for each $C(\Delta, k, l) \neq 0$ **do**
    for each state i in $S_{k\to l}$ **do**
      $E[\tau_i|s(0) = k, s(t) = l] = \frac{\sum_{m=0}^{M} \frac{t}{m+1}[\sum_{n=0}^{m}(R^n)_{ki}(R^{m-n})_{il}]Pois(m;\hat{q}t)}{P_{kl}(t)} \Rightarrow O(M^2)$
      $E[\tau_i|O, T] += C(\Delta, k, l)E[\tau_i|s(0) = k, s(t) = l]$
    end for
    for each link (i, j) in $L_{k\to l}$ **do**
      $E[n_{i,j}|s(0) = k, s(t) = l] = \frac{R_{ij}\sum_{m=1}^{M}[\sum_{n=1}^{m}(R^{n-1})_{ki}(R^{m-n})_{jl}]Pois(m;\hat{q}t)}{P_{kl}(t)} \Rightarrow O(M^2)$
      $E[n_{ij}|O, T] += C(\Delta, k, l)E[n_{i,j}|s(0) = k, s(t) = l]$
    end for
  end for
end for
Soft: $O(\hat{M}S^3 + rS^3M^2 + rS^2LM^2)$; Hard: $O(\hat{M}S^3 + min(rS^2, V)SM^2 + min(rS^2, V)LM^2)$

Algorithm 4 Eigen Algorithm

Perform eigendecomposition: $Q = UDU^{-1} \Rightarrow O(S^3)$
for $\Delta = 1$ **to** r **do**
  Compute matrix Ψ with $t = t_\Delta \Rightarrow O(S^2)$
  for each $C(\Delta, k, l) \neq 0$ **do**
    for each state i in $S_{k\to l}$ **do**
      $E[\tau_i|s(0) = k, s(t) = l] = \frac{\sum_{p=1}^{|S|} U_{kp}U_{pi}^{-1}\sum_{q=1}^{|S|} U_{iq}U_{ql}^{-1}\Psi_{pq}(t)}{P_{kl}(t)} \Rightarrow O(S^2)$
      $E[\tau_i|O, T] += C(\Delta, k, l)E[\tau_i|s(0) = k, s(t) = l]$
    end for
    for each link (i, j) in $L_{k\to l}$ **do**
      $E[n_{ij}|s(0) = k, s(t) = l] = q_{ij}\frac{\sum_{p=1}^{|S|} U_{kp}U_{pi}^{-1}\sum_{q=1}^{|S|} U_{jq}U_{ql}^{-1}\Psi_{pq}(t)}{P_{kl}(t)} \Rightarrow O(S^2)$
      $E[n_{ij}|O, T] += C(\Delta, k, l)E[n_{ij}|s(0) = k, s(t) = l]$
    end for
  end for
end for
Soft: $O(rS^5 + rLS^4)$; Hard: $O(min(rS^2, V)S^3 + min(rS^2, V)LS^2)$

(one less order of S). It is because *Expm* can efficiently compute expectations for one intermediate state/edge for all possible two end-states in a single operation $O((2S)^3)$, while *S(Eigen)* computes for every two end-states separately $O(S^4)$.

The comparison between *Expm* and *Unif* depends on the relative scale between state space S and M^2, where $M = \ulcorner 4 + 6\sqrt{\max_i\ q_i t} + (\max_i\ q_i t)\urcorner$. *Expm* is less sensitive to $\max_i q_i t$ than *Unif* method (log versus quadratic dependency), because when *Expm* is evaluated using the *scaling and squaring* method [23], the number of matrix multiplications depends on the number of doing scaling and squaring, which is $\ulcorner \log_2(||Qt||_1/\theta_{13})\urcorner$, where $\theta_{13} = 5.4$, the *Pade* approximant with degree 13, is used if scaling of Q is required [23]. Then we have $\log_2(||Qt||_1) \leq \log_2(\max_i q_i St)$. Thus, the running time of *Unif* will change according to max $q_i t$ more dramatically than *Expm* method. We believe that *Expm* is the most robust under soft EM.

Table 3: Time complexity comparison of all methods in evaluating the required expectations under Soft/Hard EM (r: number of distinct time interval, S: number of states, L: number of edges, V: number of visits, M: the truncation point for *Unif*, set as $\ulcorner 4 + 6\sqrt{\hat{q}t_\Delta} + (\hat{q}t_\Delta)\urcorner$, where $\hat{q} = max_i\ q_i$).

complexity	Expm	Unif	Eigen
Soft EM	$O(r(2S)^4 + rL(2S)^3)$	$O(\hat{M}S^3 + rS^3M^2 + rS^2LM^2)$	$O(rS^5 + rLS^4)$
Hard EM	$O(r(2S)^4 + rL(2S)^3)$	$O(\hat{M}S^3 + \min(rS^2, V)SM^2$ $+ \min(rS^2, V)LM^2)$	$O(\min(rS^2, V)S^3$ $+ \min(rS^2, V)LS^2$

3.6 Experimental results

In this section, we demonstrate the performance of the presented methods on both synthetic data and real-world dataset. The learning accuracy and running time will both be assessed. To evaluate learning accuracy of q_{ij}, we calculate the relative 2-norm error as:

$$\text{relative 2-norm error} = \frac{|\hat{q} - q|}{|q|}.$$

where $\hat{q}$ is a vector contains all learned q_{ij} parameters, and q is the ground truth. The convergence criteria is that the relative data likelihood change $\frac{|\hat{L}_i - \hat{L}_{i-1}|}{\hat{L}_{i-1}} \leq tolerace$, where the tolerance is set to 10^{-8} in most of our experiments.

In Section 3.6.1, we describe our procedure in generating the synthetic data. In

Section 3.6.2, we test the learning accuracy w.r.t. varying sampling intervals of observations using *Soft(Expm)* method. In Section 3.6.3, we test the accuracy of all learning methods under different levels of observational noises on a 5-state complete digraph. In Section 3.6.4, we test the running time performance of all methods on a 2-D gridded forwarding model of 100 states.

In Section 3.6.5, we test CT-HMM's ability in predicting future measurements for Glaucoma progression using a 2-D gridded forwarding model. Finally, in Section 3.6.6, we learn the state transition trends using a real dataset of Alzheimer's disease with a 3-D gridded forwarding model. The running time performance of all methods on both dataset will be compared. Insights are gained in visualizing the trends of disease progression, which also support the current understanding of the diseases.

3.6.1 Procedure in generating synthetic data

To produce synthetic dataset, the first step is to generate a synthetic rate matrix Q. Our procedure is listed in Algorithm 5. One can input a desired range $[h_a\ h_b]$ for the holding time parameters q_i, excluding 0 for the absorbing state. For simulation purpose, it may be better that the ratio $\frac{h_b}{h_a}$ to be not high, so that when generate synthetic data sequences given a total duration, one doesn't just stuck in one state which has large holding time, but can have more overall state jumps. We find that setting the ratio to be 5 is a reasonable choice for experiments.

After setting up Q, we will draw many realizations of the state chains and then the observations from the states. Our procedure for generating the underlying CTMCs is in Algorithm 6. Both the state and its duration are sampled based on the Q matrix. The initial state is sampled based on an input initial state distribution π. The duration τ_i for the current state i is sampled based on the exponential distribution on the holding time parameter q_i: $f(t|\mu = 1/q_i) = \frac{1}{\mu}e^{\frac{-t}{\mu}} = q_i e^{-tq_i}$. The next state is sampled based on the instantaneous transition probability: $v_{ij} = \frac{q_{ij}}{q_i}$. We sample the

Algorithm 5 Procedure: Generate the synthetic rate matrix Q

Input: State set S, edge set L, range of holding time parameter $q_i = [h_a \ h_b]$
Output: $Q = \{q_{ij}\}$
for each state i not absorbing state **do**
  Sample q_i uniformly from $[h_a \ \ h_b]$
end for
for each state i in S **do**
  for each edge (i, j) in L **do**
    Sample a random value a_{ij} uniformly from [0 1]
  end for
  Set rate q_{ij} values as: $q_{ij} = q_i \frac{a_{ij}}{\sum_{j,j\neq i} a_{ij}}$; $\quad q_{ii} = -\sum_{j,j\neq i} q_{ij}$
end for

chain until the specified total time has been reached.

Algorithm 6 Procedure: Sample a random CTMC

Input: a total time T, transition rate matrix Q, initial state distribution π
Output: a CTMC $M = \{(s_1, \tau_1), (s_2, \tau_2), ...\}$
Sample the first state i based on π
repeat
  Sample the duration τ_i for the current state i based on the exponential distribution: $f(t|\mu = 1/q_i) = \frac{1}{\mu} e^{\frac{-t}{\mu}} = q_i e^{-tq_i}$
  $T = T - \tau_i$
  if $T > 0$ **then**
    Sample the next state j based on instant transition probability $v_{ij} = q_{ij}/q_i$. Set current state $i = j$
  end if
until $T \leq 0$

After a CTMC has been realized, we generate observations with respect to a specified sampling interval $[\Delta_a \ \ \Delta_b]$. The procedure is listed in Algorithm 7. Given the last observed time t_k, the next observation time will be given by $t_{k+1} = t_k + \Delta$ where Δ is uniformly drawn from $[\Delta_a \ \ \Delta_b]$. Given the sampled time, the corresponding hidden state is first mapped, and then the observational data from this state is drawn from its data emission model $p(o|s)$. In our simulations, we use a simple Gaussian distribution $N(\mu, \sigma)$ as the data emission model.

Algorithm 7 Procedure: Sample observations from a CTMC

1: **Input:** a CTMC $M = \{(s_1, \tau_1), (s_2, \tau_2), ...\}$, observation time interval $[\Delta_a\ \Delta_b]$, data emission models $p(o|s_i)$
2: **Output:** observations
3: Set $T = M's$ total duration; set current time $t = 0$
4: **repeat**
5: Find the underlying state i at time t from M
6: Sample an observation from state i's data emission model $p(o|s_i)$
7: Sample a time interval Δ uniformly from $[\Delta_a\ \Delta_b]$
8: Set $t = t + \Delta$
9: **until** $t \geq T$

3.6.2 Simulation 1: parameter accuracy under different sampling intervals

In this experiments, we test the learning accuracy of *Soft(Expm)* method under different sampling intervals, each with the same total number of observations. The purpose of this experiment is to study the influence of sampling rate to the learning accuracy.

We set $\tau_1 = 0.5\frac{1}{max_i q_i}$ (half of the smallest mean holding time of states from Q), representing a dense sampling scenario. This sampling rate setting is inspired by the *Nyquist-Shannon sampling theorem* [94] from the field of digital signal processing, which is a a fundamental bridge between continuous-time signals ("analog signals") and discrete-time signals ("digital signals"). It establishes a sufficient condition between a signal's bandwidth and the sampling rate that allows discrete samples to capture all the information from the continuous-time one. [1] It expresses the sufficient sample rate in terms of the bandwidth for the referred class of functions: for a given sample rate f_s, perfect signal reconstruction is guaranteed possible for a band-limit function of frequency $B \leq \frac{f_s}{2}$ [94].

We thus set the sampling interval $\tau_1 = 0.5\frac{1}{max_i q_i}$ to mimic the *Nyquist-Shannon* sampling rate, set $\tau_0 = 0.5\tau_1$ for testing even denser sampling, and set $\tau_2 = 2\tau_1$ and $\tau_3 = 4\tau_1$ to test coarser sampling intervals. We test on a 5-state complete

[1] Strictly speaking, the theorem only applies to a class of functions having a *Fourier* transform that is zero outside of a finite region of frequencies [94].

Table 4: Learning errors and convergence behavior of *Soft(Expm)* method under different sampling intervals of observations. The results are averaged from 5 random runs. The sampling interval $\tau_1 = 0.5\frac{1}{max_i q_i}$ (half of the smallest mean holding time of the ground truth Q), $\tau_0 = 0.5\tau_1$, $\tau_2 = 2\tau_1$, and $\tau_3 = 4\tau_1$. For each setting, the number of observation is fixed to be $= 10^6$. Convergence criteria: relative data likelihood change $\leq tol$, where $tol = 10^{-5}$ or 10^{-8}.

tol $= 10^{-5}$	τ_0	τ_1	τ_2	τ_3
mean 2-norm err	0.0115	0.0167	0.0393	0.1679
std 2-norm err	0.0020	0.0028	0.0073	0.0157
ave num of iter	10	14	28	70
tol $= 10^{-8}$	τ_0	τ_1	τ_2	τ_3
mean 2-norm err	0.0093	0.0094	0.0103	0.0193
std 2-norm err	0.0006	0.0010	0.0073	0.0025
ave num of iter	40	79	161	620

digraph, which has 20 unknown q_{ij} parameters. The ground truth Q matrix is set to have the same holding time parameter $q_i = 1$ such that the sampling rate is equally dense/coarse for every state. The total number of observations is fixed to be $= 10^6$ for each different sampling rate.

The experimental results are listed in Table 4 by using *Soft(Expm)* method. In both convergence tolerance settings (tol $= 10^{-5}$ and 10^{-8}), the sampling rate τ_1 result in around 1% of errors though its performance is worse than the denser sampling rate τ_0. When the tolerance is set to 10^{-8}, τ_2 and τ_3 can also achieve low error rates, but the convergence speed is much slower than τ_1. As τ_1 gives good reconstruction results, we will use $\tau_1 = 0.5\frac{1}{max_i q_i}$ for the rest simulation experiments.

In Fig. **??** , we show the error rates of the *Soft(Expm)* method under different number of total observations $(10^3, 10^4, ..., 10^7)$ using $tol = 10^{-5}$.

3.6.3 Simulation 2: a 5-state complete digraph with varying noise levels

We then test the learning accuracy of all methods, *Soft(Expm), Hard(Expm), Soft(Unif), and hard(Unif), and Soft(Eigen), Hard(Eigen)*, on a 5-state complete digraph, which has a directed edge between each state pair (20 q_{ij} parameters to be estimated). The

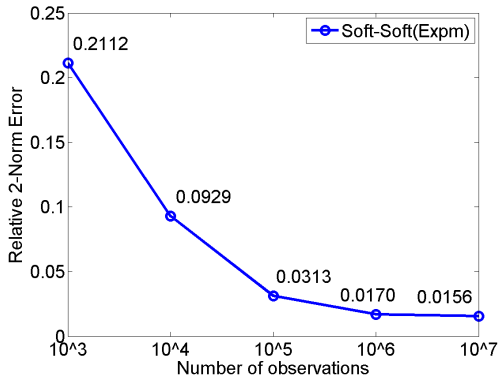


Figure 26: Relative 2-norm error w.r.t. different observation counts $(10^3, 10^4, 10^5, 10^6, 10^7)$ using *Soft(Expm)* method with $\tau_1 = 0.5\frac{1}{max_i q_i}$ sampling rate.

ground truth Q matrix is set up such that each q_i is randomly drawn from [1 5] and then q_{ij} is drawn from $[0, 1]$ and renormalized such that $\sum_{j \neq i} q_{ij} = q_i$. The state chains are generated from Q, such that each chain has a total duration around $T = \frac{100}{\min_i q_i}$, where $\frac{1}{\min_i q_i}$ is the largest mean holding time. The data emission model for state i is set as $N(i, \sigma^2)$ with $\sigma = 1/4, 3/8, 1/2, 1$, and 2 to test the performance under different noise levels. The observations are then sampled from the state chains with sampling rate $\frac{0.5}{\max_i q_i}$, where $\frac{1}{\max_i q_i}$ is the smallest mean holding time, which should be dense enough to make the chain identifiable. A total of 10^5 observations are sampled.

The simulation results from 5 random runs are listed in Table 5. All methods achieves low error rates when the noise level is low ($sigma = 1/4$). (*Eigen* fails at least once for each setting but when it works, it produces similar results as the other two). We find that all *Soft* methods achieve significantly better accuracy than *Hard* methods, especially when the noise level becomes higher, suggesting that doing soft is superior to doing hard. This can be contributed to the maintenance of the full hidden state distribution which makes it more robust to noise influence.

The convergence behavior of all methods under low noise level $sigma = 1/4$ from 2 random runs are shown in Fig. 27, where we can see that *Expm* and *Unif* method

Table 5: The average 2-norm relative error from 5 random runs on a 5-state complete digraph with varying noise levels (*sigma* in the data emission model $N(\mu, \sigma^2)$ where *mu* is set to the state index). Number of observations is 10^5. Convergence when relative data likelihood change $\leq 10^{-8}$. *Eigen* fails at least one run for each setting (but when it works, it produces similar results as the other two).

Error	$\sigma = 1/4$	$\sigma = 3/8$	$\sigma = 1/2$	$\sigma = 1$	$\sigma = 2$
S(Expm),S(Unif)	0.0261±0.0080	0.0324±0.0080	0.0420±0.0118	0.1990±0.0835	0.5096±0.1037
H(Expm),H(Unif)	0.0314±0.0089	0.1968±0.0622	0.4762±0.0995	0.8568±0.0801	0.9249±0.0298

has overlapped curves on either hard or soft setting while *Eigen* converges a bit slower but finally achieve similar results to the other two. This phenomenon is observed in all runs when *Eigen* works.

3.6.4 Simulation 3: a large 2-D forwarding model

In this experiment, we test the performance of all proposed EM methods as well as a baseline method, called Nest-Viterbi (and denoted it as H(NestV)), that we developed in [52], on a 2-D forwarding model of 100 states (10×10 grids of states). The 2-D model structure is illustrated in Fig. 28(a). The purpose of this experiment is to understand the accuracy as well as the time efficiency of each method for large state space setting with sparse edge structures.

To set up the ground truth rate matrix Q, we set the holding time parameter q_i to be in the range [1 2] except the absorbing state. The remaining parameters are set according to the procedure (Algorithm 5). We generate longitudinal sequences as follows. First, an initial state is uniformly drawn from the state space. We then sample the state chain until the chain has duration $T = \frac{10}{min_i q_i}$. The data emission model for state at position (i, j) is set as $N((i, j), (0.25, 0.25))$. We generate the observations using 50 distinct sampling interval $r = 50$, but each interval is around half of the smallest mean holding time $((\frac{1}{max_i q_i} \times 0.5) \times (1 + 0.1 rand())$.The procedure is repeated until we have a total of 5×10^5 observations.

Below we will first briefly explain the baseline method in Section 3.6.4.1, and then

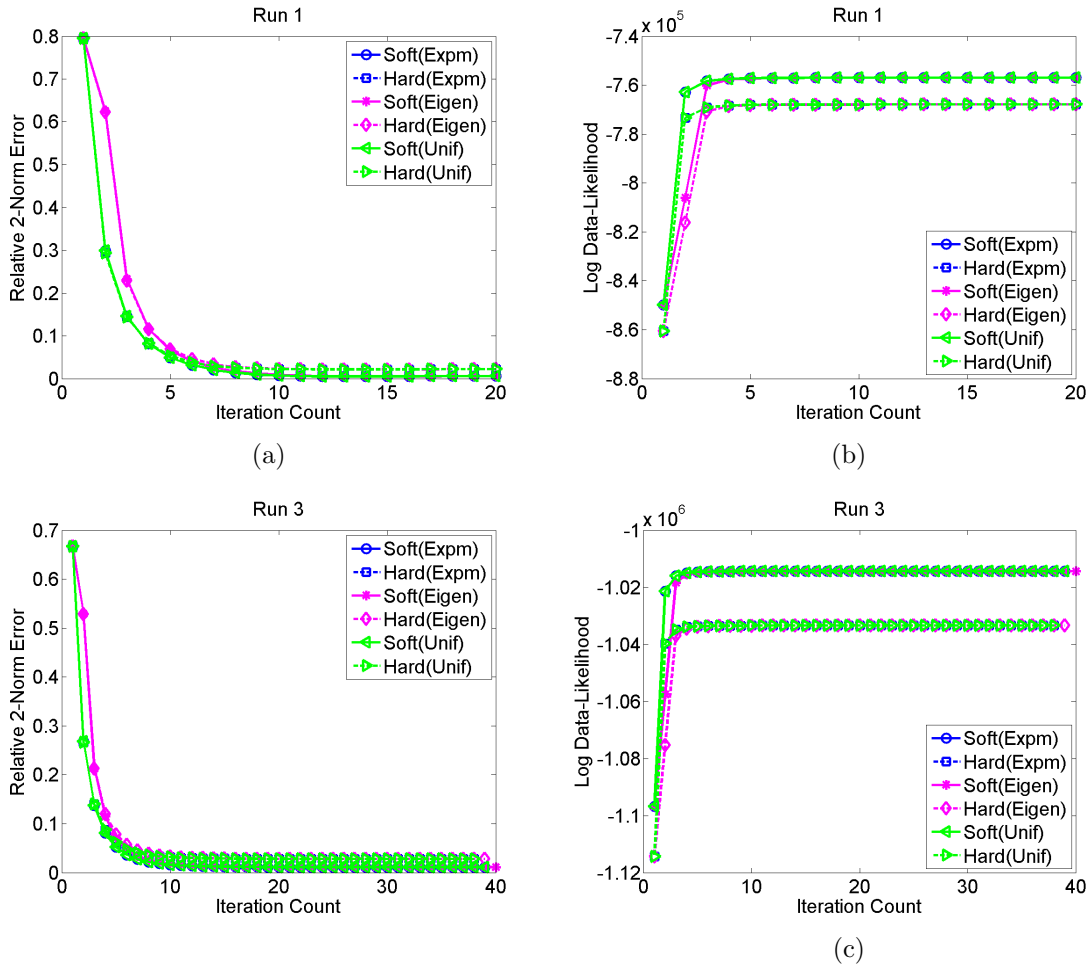


Figure 27: Convergence behavior of different learning methods on two random runs for the experiment of a 5-state complete digraph. Convergence tolerance 10^{-8}. Expm (blue line) and Unif (green line) are almost overlapped in the graph and Eigen (magenta line) method has a relatively slower convergence rate than the other two methods, possibly due to the residue error in eigen-decomposition applied on an arbitrary possibly non-diagonaliable random matrix.

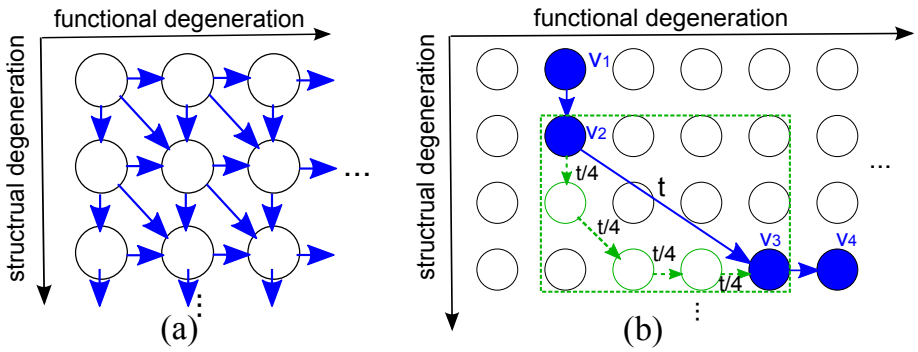


Figure 28: (a) The 2D state structure for glaucoma progression analysis [52]. The blue arrows are the allowed instantaneous transition ($q_{ij} \geq 0$). (b) An example of hidden state decoding in *Nest-Viterbi* method. The blue path represents the decoded states for the visiting data (v_1, v_2, v_3, v_4). The green path represents a possible inner state path between the two non-adjacent decoded states for visit 2 and 3. The duration t between visit 2 and 3 is uniformly distributed to the intermediate states for a coarse probability evaluation.

present the experimental results in Section 3.6.4.2.

3.6.4.1 Baseline method: Nest-Viterbi

We developed an EM algorithm, named *Nest-Viterbi EM* [52], for the multi-dimensional forwarding disease model to coarsely estimate the Q parameters, with simplified assumptions on state transition timing and duration. This method is used as a baseline for performance comparison. The main idea is to use *Viterbi* decoding to find both the outer and the inner hidden state sequence, where the outer state sequence corresponds to the actual visits, while the inner state sequence is derived by decoding of the intermediate states for any two successive outer states which are non-adjacent (no direct edge). For any two successive decoded outer states, if they are adjacent, we assume that the state transition happens exactly at the second visit. If the two states are not adjacent, we find the most probable inner state sequence with a constraint that each intermediate state occupies the same duration. This assumption is reasonable if the holding time for adjacent states is generally smoothly changed.

With predetermined transition timings, we can then easily apply *Viterbi* decoding to find the assumed most probable intermediate states. In more detail, we determine

Table 6: Performance comparison between all the methods on the 2-D forwarding model of 100 states (10×10 grids) of 297 q_{ij} parameters. Number of observations is 5×10^5. Number of distinct time intervals $r = 50$. Convergence tolerance = 10^{-5}. *Eigen* method fails in this experiment.

	S(Expm)	S(Unif)	H(Expm)	H(Unif)	H(NestV)
2-norm err	0.0587	0.0587	0.0600	0.0600	0.1491
num of iter	7	7	6	6	5
time in precomp	0 s	0.18 s	0 s	0.18 s	0.28 s
time in outer	120 min	120 min	60 min	60 min	60 min
time in inner	4.3 min	50 min	4.3 min	3.5 min	33 sec

the most probable inner path S_{kl} between state k and l, by requiring that intermediate states are distinct and adjacent with same duration (i.e., the total duration t is divided uniformly into each intermediate state) (illustrated in Fig. 28(b)). This can be formulated as: $p(S_{kl}|Q_0) = max_{S_{kl}=s_1,\ldots s_g, g\in g_{kl}, s_i\neq s_j, 1\leq i,j\leq g} \prod_{u=1}^{g-1} P_{s_u,s_{u+1}}(\frac{t}{g-1})$, where g_{kl} is the set of all possible lengths of intermediate state paths between states k and l. After deriving the single best state path for all subjects, each q_{ij} parameter can then be updated using the total transition counts through link (i, j) divided by the total duration at state i.

3.6.4.2 Results

The results are listed in Table 6. Similar to the 5-state experiments, soft approach performs better than hard approach in terms of accuracy. The H(NestV) method results in the largest error, which may be contributed to its restrictive assumptions on state transition timing and consideration of only one best state path for the actual visits and also in between two decoded outer states, rather than the full distributions or the expectations.

In terms of running time performance, it is as expected that doing soft in the outer layer has doubled time of doing hard, as soft needs both forward and backward pass to compute $p(v, s(t_v), s(t_{v+1}))$, while hard only requires a forward Viterbi

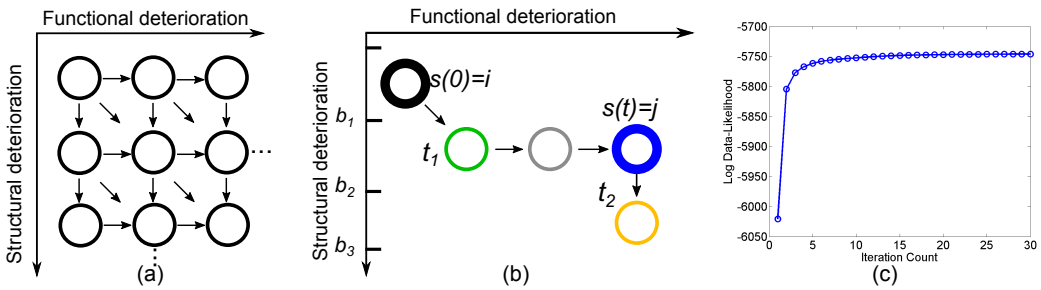


Figure 29: (a) The 2-D gridded state structure for glaucoma progression modeling. (b) Illustration of the prediction procedure. (c) Convergence behavior on the Glaucoma dataset.

decoding. When compare *S(Expm)* and *S(Unif)* in time required to compute expectations, *Expm* is more time efficient than Unif (4.3 min versus 50 min). (If we try to generate a random initial Q matrix to make *Eigen* feasible, the running time is 40 minutes in computing expectations). The better time efficiency of *S(Expm)* can be contributed to its efficiency in deriving expectations for one specified intermediate transition/duration from all possible end-state pairs, using just a single matrix exponential operation. Our results show the benefits of *Expm* in soft EM learning.

We now compare the time efficiency between *H(Expm)* and *H(Unif)*. When doing hard in the outer layer, one can expect a sparse structure in the count table $C(\Delta, k, l)$ (number of counts the data are in starting state k and in ending state l after time t_Δ). Because *Unif* can evaluate only the expectations specified by the required end-states from the best decoded paths, *Unif* can be more efficient than *Expm* in hard decoding approach. Our results in Table 6 shows that *H(Unif)* runs faster than *H(Expm)* in this experiment.

3.6.5 Real data: prediction of Glaucoma progression

We then apply CT-HMM to visualize and predict Glaucoma progression using a real Glaucoma dataset. Glaucoma is a leading cause of blindness and visual morbidity worldwide [36]. This disease is characterized by a slowly progressing optic neuropathy with associated irreversible structural and functional damage. There are conflicting

findings in the temporal ordering of detectable structural and functional changes, which confound glaucoma clinical assessment and treatment plans [97]. Here, we use a 2-D gridded state space defined by successive value bands of the two main glaucoma markers, Visual Field Index (VFI) (functional marker) and mean RNFL (Retinal Nerve Fiber Layer) thickness (structural marker) with forwarding links (see Fig. 29(a)), to understand glaucoma progression as well as to test the predictive power of the CT-HMM method.

Our glaucoma dataset contains 101 glaucomatous eyes from 74 patients followed for an average of 11.7±4.5 years, and each eye has at least 5 visits (average 7.1±3.1 visits). 63 distinct time intervals are found. The state space is created so that most states have at least 5 raw measurements mapped to it. The states which are in the straight path in between two successive raw data are instantiated, resulting in 105 states. The data emission model is set as a normal distribution with μ set to the center of the data band, and σ set to 0.25 of the band width. Ten-fold cross validation is used Testing proceeds by decoding the first 4 visits using the learned CT-HMM model and then predicting future states and observations.

In order to compare our prediction results to current state-of-the-art methods used in Glaucoma literatures which produce continuous measurements, we also design a simple procedure to generate future continuous measurements, which is described as below and illustrated in Fig. 29(b). Given a testing patient, Viterbi decoding [68] is used to decode the hidden state path for the past visits. Then, given the future time t, the most probable future state is predicted by $j = \max_j P_{ij}(t)$ (blue node), where i is the current state (black node). To predict the continuous measurements, for each disease marker separately, we search for the future time t_1 and t_2 when the patient enters a state which has same data range and next data range compared to state j, respectively. Both t_1 and t_2 can be found in a desired time resolution by binary searching. The measurement at time t can then be computed by linear interpolation

Table 7: Running time comparison for all the methods for the real glaucoma dataset (*Eigen* method fails in this experiment).

Time	S(Expm)	S(Unif)	H(Unif)
time/iter	18 min	105 min	2 min

between t_1 and t_2 and the two data bounds of state j ([$b1$ $b2$] in Fig. 29(b)).

The mean absolute error (MAE) between the predicted values and the actual measurements was used for performance assessment. The performance of CT-HMM was compared to both the conventional linear regression and *Bayesian Joint Linear Regression* [62]. For Bayesian method, the joint prior distribution of the four parameters (two intercepts and two slopes) computed from the training set [62] is used alongside with the data likelihood.

In terms of running time, *S(Expm)* spends around 18 minutes in each iteration on a 2.67 GHz machine with unoptimized MATLAB code (As a comparison, *S(Unif)* spends 105 minutes in the first iteration and becomes even slower later on; *H(Unif)* spends an average of 2 minutes per iteration, while *Eigen* fails). One fold of the convergence behavior of CT-HMM learning using *S(Expm)* is shown in Fig. 29(c).

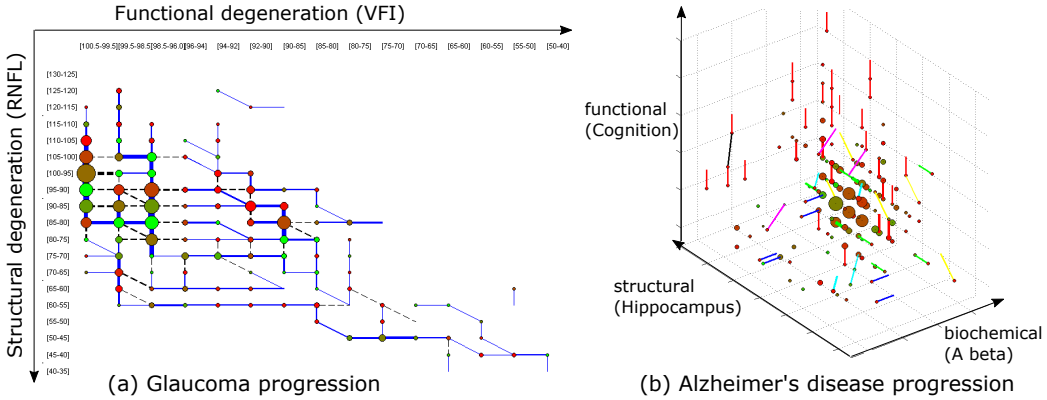
The prediction results are shown in Table 8. Our results show that CT-HMM significantly outperforms both competing methods. This may contribute to the gridded state-based structure in our 2-D CT-HMM that can flexibly capture the complex nonlinear interaction between the two markers at each combinatorial disease state. In this experiment, we find that *S(Expm)* and *H(Unif)* achieve similar performance, while *H(Unif)* can have much less running time. This suggests that for very large dataset and large state space, *H(Unif)* may be a reasonable alternative than conducting soft EM learning to trade accuracy with time.

In Fig. 30(a), we visualize the model trained using the entire dataset. Several dominant paths can be identified: there is an early disease phase with RNFL thinning with perfect vision (blue vertical path in the first column), and at around RNFL

Table 8: The mean absolute error (MAE) of predicting the two future measurements (VFI, RNFL) using our 2-D CT-HMM, Bayesian Joint Linear Regression (BJLR) [62], and the conventional linear regression (LR) method. T-test results show that our method performs significantly better than both the competing models.

MAE	H(Unif)	S(Expm)	BJLR	LR
VFI	4.51 ± 10.38	4.64 ± 10.06	5.57 ± 11.11	7.00 ± 12.22
RNFL	7.09 ± 6.49	7.05 ± 6.57	9.65 ± 8.42	18.13 ± 20.70

range $[80, 85]$ the transition trend reverses and VFI changes become more evident (blue horizontal paths). This L shape in progression supports the finding in [95] that RNFL thickness of around 77 microns acts as a tipping point at which functional deterioration became clinically observable with structural deterioration. Our 2D CT-HMM model efficiently visualize the non-linear relationship between structural and functional degeneration, which helps understanding glaucoma progression comprehensively.



(a) Glaucoma progression (b) Alzheimer's disease progression

Figure 30: Visualization scheme: (a) The strongest transition among the three instantaneous links from each state are shown in blue while other transitions are drawn in dotted black. The line width and the node size reflect the expected count. The node color represents the average sojourn time (red to green: 0 to 5 years and above). (b) The vis scheme is similar to (a) but the strongest transition link from each state is color coded as follows: $A\beta$ direction only (blue), *hippo* only (green), *cog* only (red), $A\beta$+*hippo* (cyan), $A\beta$+*cog* (magenta), *hippo*+*cog* (yellow), $A\beta$+*hippo*+*cog*(*black*). The node color represents the average sojourn time (red to green: 0 to 3 years and above).

Table 9: Running time comparison for all the methods for the Alzheimer's dataset (*Eigen* method fails).

Time	S(Expm)	S(Unif)	H(Unif)
time/iter	17 min	48 min	2 min

3.6.6 Real data: exploratory analysis on Alzheimer's disease

In this experiment we demonstrate the use of CT-HMM as an exploratory tool to visualize and understand the temporal interaction of disease markers for Alzheimer's disease (AD). AD is an irreversible neuro-degenerative disease that results in a loss of mental function due to the degeneration of brain tissues [84]. An estimated 5.3 million Americans have AD, yet no prevention methods or cures have been found [84]. It will be beneficial to understand the relationship among clinical, imaging, biochemical markers as well as genetic factors on the entire spectrum of AD as the pathology evolves, which may aid in preventing AD and developing treatments.

In this experiment, we analyze the temporal interaction among the three kinds of markers: amyloid beta ($A\beta$) level in cerebral spinal fluid (CSF) (biochemical marker), hippocampus volume (structural marker), and ADAS cognition score (functional marker) over the course of the disease. We obtained the *ADNI* (The *Alzheimers Disease Neuroimaging Initiative*)1 dataset from the website [84]. The mild cognition impairment (MCI) and AD patients who have at least two visits of all three indicated markers are included, which results in 206 subjects of 2.38 ± 0.66 visits traced in 1.56 ± 0.86 years with only 3 distinct time intervals in month resolution. A 3-D gridded state space with forwarding links is defined such that for each marker, we have 14 bands that span its value range. The procedure for constructing the state space and the definition of data emission model is the same as in the Glaucoma experiment. 277 states are instantiated and the model is then trained using *Soft(Expm)* (the running time comparison from different methods is shown in Table 9).

The 3D visualization result is shown in Fig. 30(b). The state transition trends

show that the abnormality of $A\beta$ level emerges first (blue lines) when cognition scores are still normal. Hippocampus atrophy happens more often (green lines) when $A\beta$ levels are already low and cognition has started to shown abnormality. Most cognition degeneration happens (red lines) when both $A\beta$ levels and Hippocampus volume are already in abnormal stages. Our quantitative visualization results supports the recent findings that the decreasing of $A\beta$ level in CSF is an early marker before detectable hippocampus atrophy in cognition-normal elderly [17]. Our results show that CT-HMM disease model armed with 2D/3D interactive visualization functions can be utilized as an exploratory tool to gain insights of the disease progression and generate hypothesis to be further investigated by medical researchers.

3.7 Future work: incorporation of covariate effects

In studying disease progression, it is often of interest to explore the effects of additional explanatory factors (covariates) on the disease state transition, such as age, age of the organ donor, the existence of a related disease/symptom, etc. These effects could be global (disease state independence), state-dependent, or/and time-varying. Cox proportional hazard model [57] [28, 29] can be used to relate the transition intensities $q_{ij}(t)$ and covariates $z(t)$ at time t through a generalized regression form as:

$$q_{ij}(t, z(t)) = q_{ij,b} e^{W_{ij}^T z(t)},$$

where $q_{ij,b}$ is the baseline rate parameters, and W_{ij} is the weight vector for the covariates for link (i, j). For time-varying covariates, it is usually assumed that the covariate is a step function which remains constant between two observation times. [57] also described likelihood ratio and Wald test for covariate selection and hypothesis testing.

We may estimate the baseline rate parameters $q_{ij,b}$ and the weight vector W (assume the same for all links here) for the covariate alternatively during the EM iteration. We conceive one possible method and outline it as below:

(1) Given current parameters $Q^{(00)} = \{w^{(0)}, q^{(0)}_{ij,b}\}$, compute new average $q_{ij,ave}$. Then, derive new baseline $q^{(1)}_{ij,b}$ in a closed-form using the following:

$$q_{ij,ave} = \frac{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)})}{\sum_p \sum_v E(\tau_i|O_{p,v}, Q^{(00)})} \tag{23}$$

$$\approx \frac{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)}) q_{ij,pv}}{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)})} \approx \frac{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)}) q^{(1)}_{ij,b} e^{\sum_k w^{(0)}_k c_{k,pv}}}{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)})} \tag{24}$$

Then we have the closed-form update for $q^{(1)}_{ij,b}$:

$$q^{(1)}_{ij,b} = q_{ij,ave} \frac{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)})}{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(00)}) e^{\sum_k w^{(0)}_k c_{k,pv}}} \tag{25}$$

(2) Given current parameters $Q^{(01)} = \{w^{(0)}, q^{(1)}_{ij,b}\}$, compute new average $q_{ij,ave}$.

$$q_{ij,ave} = \frac{\sum_p E(n_{ij}|O_p, Q^{(01)})}{\sum_p E(\tau_i|O_p, Q^{(01)})} \approx \frac{\sum_p E(n_{ij}|O_p, Q^{(01)}) q^{(1)}_{ij,b} e^{\sum_k w^{(1)}_k c_{k,p}}}{\sum_p E(n_{ij}|O_p, Q^{(01)})} \tag{26}$$

Then, we can compute the best $w^{(1)}$ by minimizing the approximation errors of the above equation, such as using least squared errors, and summing over all q_{ij} links. We can also weigh each link q_{ij}'s importance by their expected counts, resulting in a weighted least square optimization problem.

Let's define $f(W)$ as the follows:

$$f(W) = \sum_{i,j} \left(\frac{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(01)}) (q^{(1)}_{ij,b} e^{\sum_k w_k c_{k,pv}})}{\sum_p \sum_v E(n_{ij}|O_{pv}, Q^{(01)})} - q_{ij,ave} \right)^2 \tag{27}$$

We then find the best weight vector W using standard nonlinear optimization method, such as BFGS:

$$min_W f(W), \text{ such that } W = [w_1, ..., w_K] \geq 0. \tag{28}$$

It should be noted that alternative optimization of the two sets of parameters may result in sub-optimal solutions. The ultimate goal is to design a method that can estimate the parameters simultaneously for large-scale model, which is an important future work.

3.8 Conclusion

In this chapter, we present novel EM algorithms for CT-HMM learning leveraging recent approaches [25] in evaluating the end-state conditioned expectations for CTMC. To the best of our knowledge, we are the first to present the comprehensive framework for CT-HMM learning and utilize *Expm* and *Unif* methods in CT-HMM with time complexity analysis against *Eigen* under soft and hard EM frameworks. In our experimental results using simulation and real-datasets, we find that soft EM has higher learning accuracy than hard EM especially when the noise level is high, and *Expm* works the most efficient for soft EM approaches. In hard EM setting where one intends to trade accuracy with time efficiency, *Unif* may be more efficient than *Expm* as it can compute only the expectations specified by the end-states from the best decoded paths. *Eigen* is less general among the three methods due to its requirements of diagonalizable rate matrix. We find *Eigen* indeed fails often in our experiments.

The developed CT-HMM algorithms are also evaluated in two real datasets on Glaucoma and Alzheimer's disease with applications including progression visualization and prediction. Our learned state transition dynamics support the recent literatures about the progressing mechanism of the two diseases. Our prediction results using Glaucoma dataset outperform the state-of-the-art Bayesian joint linear regression [62] for Glaucoma progression prediction. This demonstrate the practical value of CT-HMM for longitudinal disease modeling and prediction, potentially leading to cost-effective disease management.

CHAPTER IV

CTMC END-STATE CONDITIONED OPTIMAL STATE PATH DECODING AND COMPUTATION OF EXPECTED STATE DURATION

In this chapter, we will first review the prior work that tackles optimal state sequence decoding given two end-states and a total time. We will review two possible settings for the decoding problems. One is to find the most likely state and duration sequence [66], and the other is to find the most probable state sequence considering all possible duration assignments [46]. These decoding problems have practical use, such as ... [66], and can be used in both CTMC and CT-HMM models.

For the first problem setting which inferring the most likely state and duration sequence (a trajectory which is a sequence of states as well as the exact amount of time spent in each state), it is noted recently [66] that this problem appears unsolved before. [66] then solve this problem in three settings: (1) given the starting state and a total time; (2) given both the starting state and the end state, with the total time; (3) given a set of observed states in irregularly space times, with a total time. The task is to seek the most likely trajectory that passes through the given states at the specified times. The paper [66] shows that maximum likelihood trajectory is not always well-defined if the model has a specific property (explain more later), and this property can be checked using a polynomial time procedure. When well-definedness holds, the author provides exact and dynamic programming algorithm for inferring the most likely trajectories of CTMCs under the above mentioned three settings. It is found in [66] that the most likely dwell times are often infinitessimal and non-representative of typical system behavior.

The second problem setting is to find the most probable state sequence only while marginaling out the state dwell time. This setting has probability rather than likelihood, which always exist a best solution. This problem is recently solved in [46], which guarantee to find the best solution using a path extending and pruning algorithm. The possible downside of the algorithm in [46] is that the time complexity depends in a complex way of the underlying state transition matrix Q and the specified total time. There is currently no easy time complexity analysis of the method, but it is guaranteed to terminate in finite time.

In the second problem setting, [46] finds the optimal state sequence considering all possible durations. [46] computes the probability of a feasible state sequence given a total time t by solving Kolmogorov-Chapman equations using numerical methods for ODE, and it can output the probability for a grid of time points from 0 to t, which are used in prunning the paths in the dynamic program algorithm. In this dissertation, we derive an efficient closed-form expression to compute the probability of a state sequence given a total time t in $O(n^2)$ where n is the path length, when the holding time parameters q_i in the path are distinct. Our closed-form provides an efficient method to compute the same statistics which can also be used in inference tasks using sampled-paths [22].

In addition, in the second problem setting, it will also be useful to derive the expected state duration for each state in the optimal path. Both the state sequence and the expected state durations can be used in applications that require trajectory comparison and clustering. We derive the closed-form expression for efficiently computing these statistics when q_i in the path are distinct, which has time complexity $O(n^2)$ for computing only one state, and $O(n^3)$ for all states in the path. Our closed-form solution doesn't require matrix multiplication, and is more efficient than applying other alternative methods (Expm, Unif, Eigen), which needs at least $O(n^3)$ for computing expectation for one state.

Our main contributions are summarized as follows:

- We derive a closed-form formulation to evaluate the total time conditioned state sequence probability, which has time complexity of $O(n^2)$, where n is the length of the state sequence. The direct method to compute this probability using matrix exponential requires $O(n^3)$ time.

- We derive a closed-form formulation to compute the expected state duration for each state given a state path and a total time. Our closed-form has time complexity of $O(n^2)$ to compute for just one state, and $O(n^3)$ to compute for all states along the path. The competing methods require at least $O(n^3)$ for just one state, because these method all require matrix multiplication.

This chapter is organized as follows. In Section 4.1, we review the prior work for CTMC end-state and time conditioned optimal state sequence decoding. In Section 4.2, we review the known closed-form formulation to compute the time-conditioned state sequence probability using matrix exponential, and then derive our new expression to compute the same statistics without using matrix multiplication which is more time efficient. In Section 4.3, we derive our closed-form expression to compute the path and time-conditioned expected state duration. We will compare the time complexity of our new closed-form to alternative methods using *Expm, Unif, Eigen* on an auxiliary transition matrix in computing the same statistics.

4.1 Prior work on CTMC state sequence decoding

In Section 4.1.1, we will review the work for finding the most probable state sequence considering all possible duration assignments, as this setting always has a best solution, and can is useful in real applications [46]. In Section 4.1.2, we will then review the work that finds the maximum likely state and duration sequence [66] and explain in what condition the maximum likely solution does not exist.

4.1.1 Search for maximum probability state sequences considering all continuous duration assignments

The setting in [46] is to search the maximum probability state sequence considering all possible duration assignments bounded by two end-states and total duration T. The formulation is as follows. Given a CTMC $= (S, Q)$, where S is the state set and Q is the transition matrix. Define the time-dependent probability of a state sequence $P_t(G)$ as the probability that the chain visits precisely the sequence of states in $G = (s_1, ..., s_n)$ by time t given that it starts at s_1 at time 0. Let $\tau_1, \tau_2, ..., \tau_n$ be dwell time in the states in G, define:

$$P_t(G) = (\prod_{i=1}^{n} v_{s_i,s_{i+1}}) p(\sum_{i=1}^{n-1} \tau_i \leq t \leq \sum_{i=1}^{n} \tau_i). \tag{29}$$

The goal is to evaluate $P_t(G)$ over a time interval $[0t]$, which needs in the path extending and prunning algorithm [46].

To evaluate $P_t(G)$, [46] used the Chapman-Kolmogorov equations, resulting in:

$$\frac{dP_t(G)}{dt} = q_{s_{n-1}} v_{s_{n-1},s_n} P_t(\bar{G}) - q_{s_n} P_t(G) \tag{30}$$

where $\bar{G}$ is the state sequence $(s_1, ..., s_{n-1}$, that is, one-step shorter sequence of the states in G. To evaluate $P_t(G)$, one needs to solve the above linear differential equation. This equation depends on $P_t(\bar{G})$, which obeys its own linear differential equation, depending in part on the probability of a state sequence which is yet one step shorter. Thus, one can find $P_t(G)$ by solving system of linear differential equations, where the variables are $P_t((s_1))$, $P_t((s_1, s_2))$, ..., $P_t((s_1, s_2, ..., s_n))$. The initial conditions are $P_t((s_1)) = 1$ and $P_t((s_1, ..., s_i)) = 0, i > 0$. One can use standard numerical methods to solve differential equations to calculate $P_t(G)$.

Then, to find the optimal state sequence given a total time t_{max}, [46] proposed a way to guide the path extending and prunning process using a notion called *dominance*. Suppose G_1 and G_2 are two different state sequences with same starting and end state. Then G_1 dominates G_2 if $P_t(G_1) > P_t(G_2)$ for all $t \in (0, t_{max})$. If G is

not dominated by any other sequence, then G is said non-dominated. [46] proves that if the given CTMC has a finite state set, then there must be a finite number of non-dominated state sequences given a total time t_{max}. Then, if the program can list them all, and check which has the largest probability, then the maximum probability sequence is guaranteed to be found. [46] also shows that all prefixes of any non-dominated sequence must also be non-dominated.

[46] proposes a search and prunning algorithm that enumerates non-dominated sequences from shorter to longer, starting from the specified starting states. After enumerating, each non-dominated sequence is evaluated to see which is the most probable one at time t_{max}. Please see [46] for the detailed algorithm.

4.1.2 Search for maximum likelihood state and duration sequences

In [66], a trajectory of the CTMC is a sequence of states along with the dwell time in all but the last state: $U = (s_0, \tau_0, s_1, \tau_1, ..., s_{k-1}, \tau_{k-1}, s_k)$. This represents that the system enters state s_0 at the beginning and where it stays for τ_0 time, then goes to s_1, and stays for τ_1 and so on. Eventually, the system hits state s_k and remains there. Let $U_t = (s_0, \tau_0, s_1, \tau_1, ..., s_{k_t-1}, \tau_{k_t-1}, s_{k_t})$ be a random variable describing the trajectory of the system up until time t. This represents that there are k_t state transitions up until time t, where k_t itself is also a random variable.

Given the initial state s, a total time t, the likelihood of a particular trajectory U is shown as cite:

$$L(U_t = U | s_0 = s) = \begin{cases} 0 & \text{if } s_0 \neq s \text{ or } \sum_{i=0}^{k-1} t_i > t \\ (\prod_{i=0}^{k-1} q_{s_i} e^{-q_{s_i} t_i} \upsilon_{s_i, s_{i+1}})(e^{-q_{s_k}(t - \sum_i \tau_i)}) & \text{otherwise} \end{cases}$$

The condition $\sum_{i=0}^{k-1} t_i > t$ in the first line means that the likelihood is zero if the chain has total time larger than t. Otherwise, in the second case the first parenthesis $(\prod_{i=0}^{k-1} q_{s_i} e^{-q_{s_i} t_i} \upsilon_{s_i, s_{i+1}})$ represents the likelihood of the dwell time until state s_{k-1} and the state transitions in the sequence. The second parentheses $(e^{-q_{s_k}(t - \sum_i \tau_i)}$ accounts

for the probability that the dwell time in the last state does not finish before time t.

The end-conditioned most likely trajectory problem can be formulated as:

$$arg\ max_U \quad L(U_t = U | s(0) = s, s(t) = s^{'})$$

where s and $s^{'}$ are the two given end-states, and t is the given total time.

- **Find the most likelihood dwell time for a given state sequence:** In this setting, the optimization problem becomes:

$$arg\ max_{(\tau_0,...,\tau_{t_{k-1}})} \quad L(U_t = U | s_0, s_1, ..., s_k) \quad \text{s.t.} \quad \sum_{i}^{k-1} t_i < t$$

  In [66], it is shown that when given a particular state sequence, if state s_i has the largest expected dwell time (i.e., has the smallest holding time parameter q_i), then the most likely setting of dwell time is derived by putting all of the time T in state s_i (the slowest state), and all other transitions happen instantaneously. This result is not unintuitive, but is dissatisfying in the sense that the resulting most likely set of dwell time is not typical, since none are close to their expected value.

- **Find the most likely state sequence for give end-states and total time:** With the solution for the above problem that all the times goes into the slowest state, one can now finds the most likely state sequence that maximize the likelihood using the derived formula:

$$arg\ max_{(\tau_0,...,\tau_{k-1})} \quad L(U_t = U | s_0 = s(0)) = (\prod_{i=0}^{k-1} q_{s_i} e^{-q_{s_i} t_i} v_{s_i, s_{i+1}}) e^{(min_{i=0}^{k} q_{s_i})t}.$$

  The solution can be found using dynamic programming. In order to build the maximum likelihood paths of increasing length, one finds the best ways of extending the shorter paths [66]. The main difference here to other typical dynamic program problems is to remember not just the current score, the current

end-state, but also the smallest holding time parameter q along the path. The running time of this algorithm is $O(K|S|^3)$, where K is the number of state jumps, which can be limited to be a preset value.

- **Situations where maximum likelihood trajectory is not well-defined:** It is analyzed in [66] that if a CTMC has a cycle of states $(s_0, s1, ..., s_k = s_0)$ such that

$$\prod_{i=0}^{k-1} v_{s_i,s_{i+1}} q_{s_i} \geq 1$$

, than maximum likelihood trajectories do not exist. A sequence of trajectories with ever-increasing likelihood can be found if the cycle is reachable. Thus, before seeking the maximum likelihood trajectories from a starting state s, one should check if the graph has a cycle having the stated property. This can be done by setting the weight as $log v_{s_1,s_2} q_{s_1}$ for the edge s_1 to s_2, and check whether the graph contains a positive-weight cycle, which needs polynomial time computations.

4.2 Computation of time-conditioned state sequence probability

Here, the goal is to compute the probability of a state sequence $G = (s_1, ..., s_n)$ given the total spanning time t. This can be formulated as:

$$p(G = (s_1, ..., s_n)|t) = (\prod_{i=1}^{n-1} v_{s_i,s_{i+1}}) p(\sum_{i=1}^{n-1} \tau_i < t \leq \sum_{i=1}^{n} \tau_i | G = (s_1, ..., s_n)) \qquad (31)$$

where n is the length of G, $v_{s_i,s_{i+1}}$ is the state transition probability, and τ_i is the duration in each state s_i.

Note that $p(\sum_{i=1}^{n-1} \tau_i \leq t \leq \sum_{i=1}^{n} \tau_i | G)$ can be evaluated using matrix exponential on an auxiliary $\hat{Q}$ matrix constructed using only the states along the state path (Eqn. (4)(5) in [22]). In detail, if $G = (s_1, \cdots, s_n)$, we construct an auxiliary $(n+1) \times (n+1)$

rate matrix $\hat{Q}$ as follows:

$$\hat{Q} = \begin{bmatrix} -q_{s_1} & q_{s_1} & 0 & \cdots & 0 & 0 \\ 0 & -q_{s_2} & q_{s_2} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -q_(s_n) & q_{s_n} \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}_{(n+1)\times(n+1)} \tag{32}$$

where $q_{s_k} = \sum_{i,i\neq k} q_{s_k,s_i}$ is the holding time parameter of state s_k. This matrix has the structure that only the transition from s_i to s_{i+1} is set positive at the $(i, i+1)$ entry in $\hat{Q}$. It has be shown in [22] that

$$p(\sum_{i=1}^{n-1} \tau_i \leq t \leq \sum_{i=1}^{n} \tau_i | G) = (e^{\hat{Q}t})_{1,n} \tag{33}$$

which is a matrix exponential operation on the auxiliary matrix $\hat{Q}$.

Closed-form formulation for the $(1, n)$ entry of $e^{\hat{Q}t}$: We find that the $(1, n)$th entry of $e^{\hat{Q}t}$ has closed-form via *Laplace transformation*[1]. The results are as below (For notation simplicity, we write q_{s_i} as q_i):

- If $q_i, i = 1, ..., n$ are distinct, we have

$$(e^{\hat{Q}t})_{1,n} = \frac{1}{q_n} \sum_{i=1}^{n} [(\prod_{j=1, j\neq i}^{n} \frac{q_j}{q_j - q_i}) q_i e^{-q_i t}]$$

- If $q_i, i = 1, ..., n$ are not distinct, without loss of generality, we assume $q_1 = q_2 = ... = q_p$, and $q_{p+1} \neq q_{p+2} ... \neq q_n$, then we have the form below:

$$(e^{\hat{Q}t})_{1,n} = (\frac{a_{11}}{(p-1)!} t^{p-1} + \frac{a_{12}}{(p-2)!} t^{p-2} + ... + a_{1p}) e^{-q_1 t} + A_{p+1} e^{-q_{p+1} t} + ... + A_n e^{-q_n t}$$

Below we list some steps to derive $(e^{\hat{Q}t})_{1,n}$ when q_i are distinct using *Laplace Transform* and the *Inverse Laplace Transform* by *Partial Fraction Expansion* as below:

$$(e^{\hat{Q}t})_{1,n} = \mathbb{F}^{-1}(\mathbb{F}((e^{\hat{Q}t})_{1,n}))$$

[1]This is a collaborative work with Shuang Li and Le Song, who derive this exact closed-form.

where $\mathbb{F}^{-1}$ is *Laplace Transform* and $\mathbb{F}^{-1}$ is *Inverse Laplace Transform.*

$$\mathbb{F}(s) = \mathbb{F}(e^{\hat{Q}t})_{1,n} = (sI - \hat{Q})^{-1}_{1,n} = \frac{q_1 q_2 ... q_{n-1}}{(s+q_1)(s+q_2)...(s+q_n)}$$

If all $q_i, i = 1, ..., n$ are distinct, then we have

$$\mathbb{F}(s) = \frac{A_1}{(s+q_1)} + \frac{A_1}{(s+q_2)} + ... + \frac{A_{n-1}}{(s+q_{n-1})} + \frac{A_n}{(s+q_n)} (Partial\ Fraction\ Expansion)$$

, where

$$A_i = lim_{s \to -q_i}(s+q_i)\mathbb{F}(s) = \frac{q_1 q_2 ... q_{n-1}}{(-q_i+q_1)...(-q_i+q_{i-1})(-q_i+q_{i+1})...(-q_i+q_n)}.$$

Then we have

$$\begin{aligned}(e^{\hat{Q}t})_{1,n} &= \mathbb{F}^{-1}(\mathbb{F}(s)) = A_1 e^{-q_1 t} + A_2 e^{-q_2 t} + ... + A_n e^{-q_n t} \\ &= \frac{1}{q_n} \sum_{i=1}^{n} [(\prod_{j=1, j \neq i}^{n} \frac{q_j}{q_j - q_i}) q_i e^{-q_i t}]\end{aligned}$$

4.2.1 Comparison of time complexity

Here, we compare the time complexity of computing the state sequence probability given a total using a direct matrix exponential method and the closed-form expression we derived. The comparison is listed in Table 10. The direct method means that to compute $(e^{\hat{Q}t})_{1,n}$ by computing the matrix exponential and then read the $(1, n)$ entry, which needs $O(n^3)$ time. Clearly, our closed-form results only $O(n^2)$ time, but it need that all q_i are distinct.

Table 10: Time complexity comparison of all methods in evaluating time-conditioned path probability (n: number of states in the path)

Complexity	direct-expm	Ours
Time	$O(n^3)$	$O(n^2)$
Prerequisite	none	q_i distinct

4.3 Computation of path-and-time conditioned expected state duration

To the best of our knowledge, we have not found any literature that explicitly write down the closed-form and methods for computing expected state duration for a *path-conditioned* CTMC. We derive a closed-form formulation for computing these path-conditioned statistics, which is applicable when q_i for the states in the path are distinct. We also compare the time complexity of using our closed-form to using the three alternative methods (Expm, Unif, Eigen) which are originally used in end-state conditioned cases, to compute path-conditioned statistics by applying on an auxiliary matrix according to the state path.

We now explain our derived closed-form in detail[2]. Given a state sequence $G = (s_1, ..., s_n)$, a total duration t, the computation of the expected state duration τ_k, for state $k = 1, ..., n$ is as below:

$$E(\tau_k|G, \sum_{i=1}^{n-1}\tau_i \le t < \sum_{i=1}^{n}\tau_i) = E[\int_0^t \mathbf{1}(S_u = s_k)du|G, \sum_{i=1}^{n-1}\tau_i \le t < \sum_{i=1}^{n}\tau_i] \tag{34}$$

$$= \int_0^t p(S_u = s_k|G, \sum_{i=1}^{n-1}\tau_i \le t < \sum_{i=1}^{n}\tau_i)du \tag{35}$$

$$= \frac{\int_0^t (e^{\tilde{Q}_{k,A}u})_{1,k}(e^{\tilde{Q}_{k,B}(t-u)})_{1,n-k+1}du}{(e^{\tilde{Q}t})_{1,n}} \tag{36}$$

$$= \frac{\int_0^t [a_{k,1}e^{-q_1u} + \cdots + a_{k,k}e^{-q_ku}] \cdot [b_{k,k}e^{-q_k(t-u)} + \cdots + b_{k,n}e^{-q_n(t-u)}]du}{(e^{\tilde{Q}t})_{1,n}} \tag{37}$$

$$= \frac{(\sum_{i=1}^{k}\sum_{j=k,i\neq j}^{n} a_{k,i}b_{k,j}\frac{e^{-q_it}-e^{-q_jt}}{-q_i+q_j}) + t(a_{k,k}b_{k,k}e^{-q_kt})}{\frac{1}{q_n}\sum_{i=1}^{n}\left[(\prod_{j=1,j\neq i}^{n}\frac{q_j}{q_j-q_i})q_ie^{-q_it}\right]} \tag{38}$$

where $q_i = \sum_{i\neq j} q_{ij}$ are the holding time parameters, $\tilde{Q}$ is the auxiliary matrix for

[2]This is a collaborative work with Shuang Li and Le Song.

the entire state path:

$$\tilde{Q} = \begin{bmatrix} -q_1 & q_1 & 0 & \cdots & 0 & 0 \\ 0 & -q_2 & q_2 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -q_n & q_n \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}_{(n+1)\times(n+1)} , \tag{39}$$

, $\tilde{Q}_{k,A}$ is the auxiliary matrix for the partial state path from the first state to the kth state,

$$\tilde{Q}_{k,A} = \begin{bmatrix} -q_1 & q_1 & 0 & \cdots & 0 & 0 \\ 0 & -q_2 & q_2 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & q_k & q_k \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}_{(k+1)\times(k+1)} ,$$

, and $\tilde{Q}_{k,B}$ is the auxiliary matrix for the partial state path from the $(k+1)$th state to the last state (the nth state):

$$\tilde{Q}_{k,B} = \begin{bmatrix} -q_k & q_k & 0 & \cdots & 0 & 0 \\ 0 & -q_{k+1} & q_{k+1} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -q_n & q_n \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}_{(n-k+2)\times(n-k+2)}$$

We also have the closed-form expression for $(e^{\tilde{Q}t})_{1,n}$ derived from the previous section as follows:

$$(e^{\tilde{Q}t})_{1,n} = \sum_{i=1}^{n} \left[\left(\prod_{j=1, j\neq i}^{n} \frac{q_j}{q_j - q_i} \right) \frac{q_i}{q_n} e^{-q_i T} \right] = \sum_{i=1}^{n} c_i e^{-q_i T} \tag{40}$$

where $c_i = (\prod_{j=1,j\neq i}^{n} \frac{q_j}{q_j - q_i})\frac{q_i}{q_n}$. Finally, the terms $a_{k,i}$ and $b_{k,j}$ are defined as:

$$\begin{cases} a_{k,i} = \frac{q_i}{q_k}(\prod_{m=1,m\neq i}^{k} \frac{q_m}{q_m - q_i}) & k = 1, \cdots, n;\ i = 1, \cdots, k \\ b_{k,j} = \frac{q_j}{q_n}(\prod_{m=k,m\neq j}^{n} \frac{q_m}{q_m - q_j}) & k = 1, \cdots, n;\ j = k, \cdots, n \end{cases} \tag{41}$$

which are derived from the closed-form expression for $(e^{\tilde{Q}t})_{1,n}$.

Efficient computation of the expected duration for each state in the state path: we can derive the recursive relation between $a_{k,i}$ and $a_{k-1,i}$, and also $b_{k,j}$ and $b_{k+1,j}$, so that the computation of these coefficients can be efficient:

$$\begin{cases} a_{k,i} = a_{k-1,i}\frac{q_{k-1}}{q_k - q_i} & k = 1, \cdots, n;\ i = 1, \cdots, k \\ b_{k,j} = b_{k+1,j}\frac{q_k}{q_k - q_j} & k = 1, \cdots, n;\ j = k, \cdots, n \end{cases} \tag{42}$$

Time complexity analysis: The time complexity in computing all τ_k, $k = 1, ..., n$, where n is the length of the inner path is $O(n^3/6)$. The detailed analysis is as follows. First, the computation of all required $a_{k,i}$ and $b_{k,j}$ from the derived recursive form needs $O(n^2)$. Second, the computation of the common denumerator requires $O(n^2)$. Third, the computation of the numerator for one τ_k using the precomputed $a_{k,i}$ and $b_{k,j}$, requires $k(n-k) = O(n^2)$. Finally, the computation of the numerator for all τ_k needs $\sum_{k=1}^{n} k(n-k) = (n^3 - n)/6 = O(n^3/6)$.

Thus, evaluating just one expected state duration requires $O(n^2)$ computations, and evaluating all expected state durations along the path needs $O(n^3)$ (n is the path length).

4.3.1 Comparison of time complexity

We compare the time complexity of computing the expected state durations given the state path and a total time from the three alternative methods (Expm, Unif, Eigen) and our closed-form solution. The three alternative methods *(Expm, Unif, Eigen)* are originally used in computing the end-state conditioned statistics (see more details in Chapter 3). To use them for computing path-conditioned statistics, we only need

to construct the auxiliary matrix Q using the state path as the new rate matrix, and the remaining formulations are the same for calculating the end-state and path conditioned statistics.

The time complexity comparison is listed in Table 11. Our closed-form expression have only $O(n^2)$ time in computing one expected state duration, while all other methods require at least $O(n^3)$ as these alternative methods all involve matrix multiplications. When computing the expected duration for all states along the path, our close-form method has $O(n^3)$ complexity, and is more efficient than *Expm and Unif* methods. However, our method and *Eigen* both have prerequisite of its use, while *Expm* and *Unif* are more general.

Table 11: Time complexity comparison of all methods in evaluating path-and-time conditioned expected state durations. (n: number of states in the path, M: the truncation point for *Unif*, set as $\ulcorner 4 + 6\sqrt{\hat{q}t} + (\hat{q}t)\urcorner$, where $\hat{q} = max_i\ q_i$ for states in the path, and t: the total time).

Complexity	Expm	Unif	Eigen	Our Closed-Form
For one state	$O((2n)^3)$	$O(Mn^3 + M^2)$	$O(n^3 + n^2)$	$O(n^2)$
For all states	$O(n(2n)^3) = O(n^4)$	$O(Mn^3 + M^2n)$	$O(n^3 + n^3)$	$O(n^3)$
Prerequisite	none	none	Q diagonalizable	q_i distinct

4.4 An simulation example

We now show one simulation example of computing the optimal state sequence given two end-states and a total time using the method from [46] (Matlab and R package can be downloaded from the author's website), and then computing the expected state duration for each state in the optimal path using our presented closed-form and the three alternative methods (*Expm, Unif, Eigen*).

In Fig. 31, we show a simple 2-state complete digraph models, with state s_1 has $q_1 = 1$ (mean holding time = 1) and s_2 has $q_2 = 0.5$ (mean holding time = 2). Now if we set the beginning state to be s_1, the terminating state to be s_2 and a total duration = 12. What is the optimal state sequence considering all possible dwelling

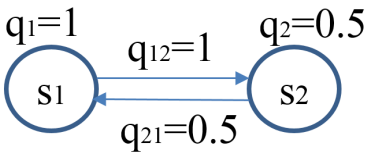


Figure 31: A simple 2-state digraph used to demonstrate the optimal state sequence decoding and computation of the expected state duration for the optimal path.

time assignments?

The path extending and pruning method from [46] gives the solution of the best state sequence as $(1, 2, 1, 2, 1, 2, 1, 2)$, and our methods to compute expected state duration given the path and total time, gives the corresponding durations as $(1, 2, 1, 2, 1, 2, 1, 2)$ which sums to 12. This result corresponds to our intuition of the best state path and the expected dwelling time of this model based on the holding time parameters q_i.

4.5 Experimental results in computing path-and-time conditioned expected state duration

We test the running time of computing path-and-time conditioned expected state duration. Our setting is as follows. Given a state path of length n, each state's holding time parameter q_i is generated randomly from [0 1]. The testing total time T is set to be the sum of the mean holding time parameter $\sum_{i=1}^{n} 1/q_i$. We test the running time for $n = 20, 30, 40$, and for each testing, 10 runs are tested and the results are averaged.

The running time results are listed in Table 12. We find that *Eigen* method tends to be the faster but fails often for n is large. Our closed-form has better running time than Expm and Unif method in all testing cases. Unif method becomes slower when the total time T is larger (when n is large). This is due to its quadratic dependency of time. In terms of the accuracy results, we find that in our testing cases, all methods can derive similar expected state duration and the durations are sum to the total time, except the runs when Eigen method fails. However, note that our closed-form

can only be used when each q_i are distinct. Regarding to the generalizability and running time together, *Expm* method appears to be the best choice.

Table 12: Running time comparison of all methods in evaluating path-and-time conditioned expected state durations. For each setting, 10 random runs are tested and the average running time is reported.

Time	Expm	Unif	Eigen	Our Closed-Form
$n = 20$	0.011 s	0.375 s	0.002 s	0.005 s
$n = 30$	0.029 s	0.876 s	0.007 s (fail: 1/10)	0.016 s
$n = 40$	0.061 s	18.91 s	0.012 s (fail: 5/10)	0.030 s

4.6 Conclusion and future work

In this chapter, we review the prior works for CTMC and CT-HMM in finding the optimal state sequence considering all possible duration assignments given two fixed end-states and a total time. Another related problem is to find the maximum likely state and duration sequence together under same conditions. It is found that the former problem setting can result in more biological meaningful solution, while the solution derived from the latter setting gives all duration to the state with the largest mean holding time (smallest q_i), which is usually undesirable in practice. The exact solution found for the first problem can be useful in real applications to understand the best hidden path. However, the prior work didn't present the method to compute the expected duration for the found optimal state path.

To the best of our knowledge, we are the first to explicitly tackle the problem of computing the path-and-time-conditioned expected state durations. While the techniques for finding end-state conditioned expectations can be applied for the path-conditioned one, by working on the auxiliary matrix constructed for the path, the resulting computation may not be the most efficient. We derive the exact closed-form for computing the following two statistics without using any matrix multiplications but with the prerequisite that q_i in the paths are all distinct. The two statistics are (1) the probability of a state sequence given a total time, computed in $O(n^2)$, (2) the

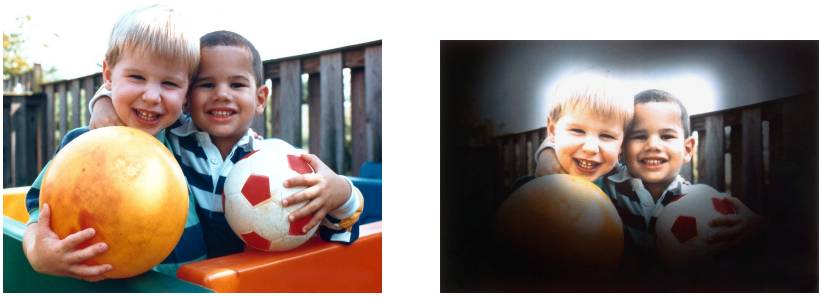
expected state duration given a state sequence and a total time, computed in $O(n^2)$ for just one state, and $O(n^3)$ for all states along the path. We also compare the time complexity of using *Expm, Unif, Eigen* methods to the newly-derived closed-form. The former three methods require at least $O(n^3)$ for just one state, because these methods all require matrix multiplication. The derived new closed-form expression, which is more time efficient, can be useful in CTMC and CT-HMM decoding and inference tasks.

CHAPTER V

APPLICATIONS ON GLAUCOMA PROGRESSION MODELING USING MULTI-DIMENSIONAL CT-HMM

Glaucoma is an optic neuropathy characterized by slowly progressive loss of retinal ganglion cells and damage of optic nerve. If left untreated, glaucoma may cause irreversible visual field deficits and even blindness (see Fig. 32) [39], and thus timely detection of disease and its progression are of paramount importance for initiating or modifying treatment. Glaucoma has been called the silent thief of sight because the loss of vision often occurs gradually without patients awareness until the disease has advanced significantly. It is estimated that over 2.2 million Americans have glaucoma but only half of those know they have it [85]. This large undiagnosed population is mainly due to unnoticeable symptoms of early glaucoma and the difficulty of consistently screening the entire population. In the U.S., more than 120,000 people are blind from glaucoma [67]. Worldwide, glaucoma is the second-leading cause of blindness after cataracts [36].

Since the visual field cannot typically be recovered, early identification of glaucoma



(a) Normal vision (b) Glaucoma

Figure 32: Comparison of normal vision and visual field with Glaucoma (From [85], used without permission).

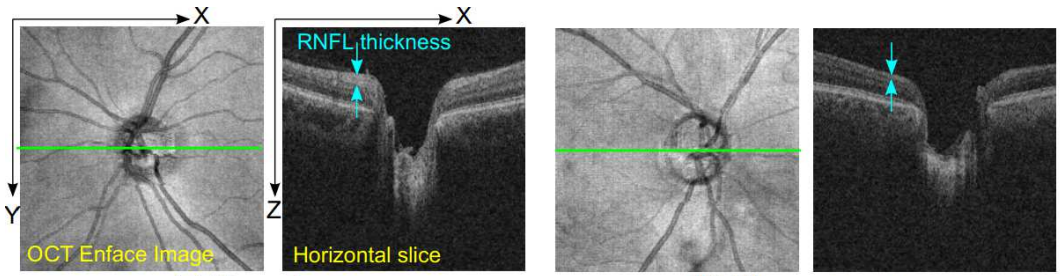


Figure 33: Examples of Optical Coherent Tomography (OCT) Images of optic nerve head. The OCT enface (left image) is a 2D image generated by projecting the 3D-OCT volume along the z (depth) axis. The horizontal slice (right image) corresponds to the green line in the enface. In the glaucoma case (b), the retinal nerve fiber layer (RNFL), the topmost layer of the retina that looks bright in OCT images, is apparently thinner than the normal case (a).

and its progression, and delivery of appropriate treatment are critical to retard the deterioration and preserve sight. Clinically, several techniques for structural and functional measurement are often utilized for glaucoma monitoring. For example, 3D optical coherence tomography (3D-OCT) is used to examine the optic nerve head [43] (Fig. 33), and psychophysical techniques, such as automated perimetry, are applied to assess the status of the visual field [40].

Many cross-sectional studies of Glaucoma progression have demonstrated good correspondence between structure and function and the ability to detect the disease with high accuracy [11] [81] [98] [96] [87] [21] [79] [15]. However, longitudinal studies have shown discordance between structural and functional changes appearing over time, regardless of the method employed for assessing structural change [44] [59] [61] [78] [45] [97] [60]. In many of the latter studies, structural changes were detected prior to functional loss in a substantial percentage of eyes, while a smaller percentage of eyes showed functional progression without detectable structural changes. This asymmetry and asynchrony between the structural and functional progression poses challenges for clinical assessment, resulting in inefficient treatment.

A number of prior studies tried to establish models of the relationship between

structural and functional factors, [1] [2] [71] [3] [12]. However, none of them has reached the level of accuracy and efficiency to be usable by clinicians to facilitate the development of treatment plans. This is most likely because conventional glaucoma progression analysis uses models which are based on restrictive assumptions, such as linearity of change. With such restrictions, various potential glaucoma progression phenotypes may be recognized as noise leading to confusing or sometimes contradictory observations.

In this dissertation, we propose to combine longitudinal structural and functional measurements by using a 2-D CT-HMM (see Fig. 34) to decode the underlying true disease stages. We use two hidden state dimensions, one is structural and the other is functional, to model the composite disease status. Our 2-D state-based model can reveal the intricate interactions between structural and functional damages without restrictive assumptions on the form of progression, such as linearity used in conventional models, and provides a useful starting point for developing a practical tool that can identify previously undiscovered glaucoma progression phenotypes and provide more accurate predictions of glaucoma progression.

This chapter is organized as follows. In Section 5.1, we train and visualize separate 2-D CT-HMMs for patient visit data from all ages, ages less than 70, and ages greater than 70, using a large glaucoma longitudinal dataset. The state transition trends are visualized and compared between these three models. We will show that our results support a recent literature for first detectable functional changes in a particular structural state. In Section 5.2, we test the ability of our 2-D CT-HMM for predicting future progression of glaucoma. A simple procedure is devised to predict not only the future state but also the continuous measurements. We compare the predictive results to the state-of-the-art methods and get promising results.

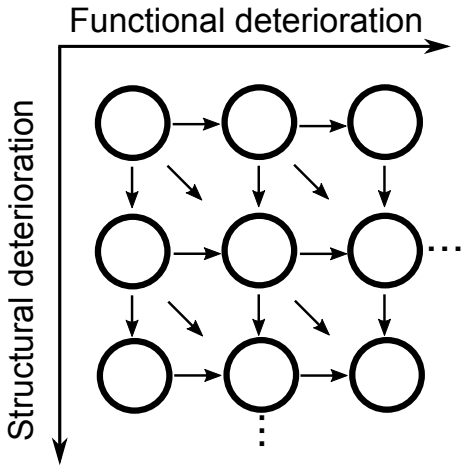


Figure 34: The 2-D state structure for glaucoma progression, where one dimension represents functional degeneration, and the other is for structure. The blue arrows are the allowed instantaneous forward transition ($q_{ij} \geq 0$).

5.1 2-D exploratory analysis using structural and functional markers

The prevailing approach currently employed assesses glaucomatous changes occurring either by structure or by function. Here, we first introduce a novel comprehensive approach based on two-dimensional CT-HMM model, where one dimension is functional degeneration, and the other dimension is structural degeneration, to model the disease states and their transition behavior considering both components.

To define the state structure for glaucoma progression, we assume that the disease can only degenerate and cannot recover since glaucomatous damage is typically irreversible. We also require that the instantaneous transitions (q_{ij}) must go to the closest degenerative states only, which is a reasonable setting since q_{ij} represents instantaneous transition. The three allowed transition links (the right, down, diagonal link) from each state and the resulting model structure are illustrated in Fig. 34. Note that it is also feasible to add recovering links (backward links) from each state if desired.

Construction of MD state space and the state transition structure: We now explain our procedure in constructing an M-D state space and the state transition

structure. Given M kinds of measures for building an M dimensional disease model, we define a disease state as having a predetermined value range for each measure. A simple histogram analysis for each measure can be used to define an appropriate value range such that most of the states have a sufficient amount of patient data associated with them. After defining the state space, we then create the states and their transition structure based on the dataset at hand, such that there exists a straight continuous path between every two consecutive input data. In this way, a compact and descriptive state space can be constructed. Note that the size of the model varies with both the number of dimensions (M-D) and the total number of disease states (which depends upon the number of states per dimension). From the computational perspective, the total number of states is the primary quantity that determines the running time of our program, and we describe below methods which can be efficient for different numbers of states. The number of dimensions (measures) used in the model will be chosen based on the clinical or research purpose.

5.1.1 Experimental results: transition trend visualization

For this experimental setting, 197 glaucoma suspect and glaucoma subjects were followed longitudinally for an average of 10.4 ± 4.9 years, with comprehensive ocular examination, visual field (VF) testing and optical coherence tomography (OCT). The total number of visits with qualified VF and OCT measurements is 1048. Through the course of the study, multiple OCT iterations were used. Calibration equations between four OCT machines were employed to enable the use of mean retinal nerve fiber layer (RNFL) thickness measurements as a one-parameter continuum. 2D state space defined by using VF index (VFI) and mean RNFL is employed to evaluate progression.

In Fig. 35, the transition results learned from all the data are shown. The strongest

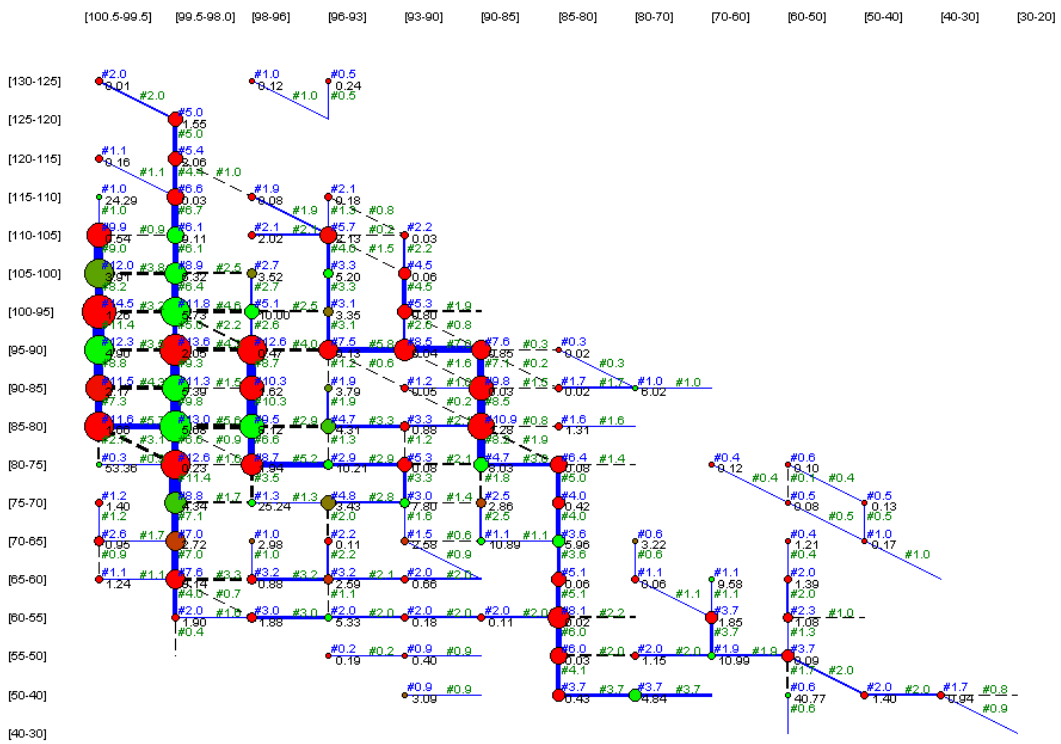
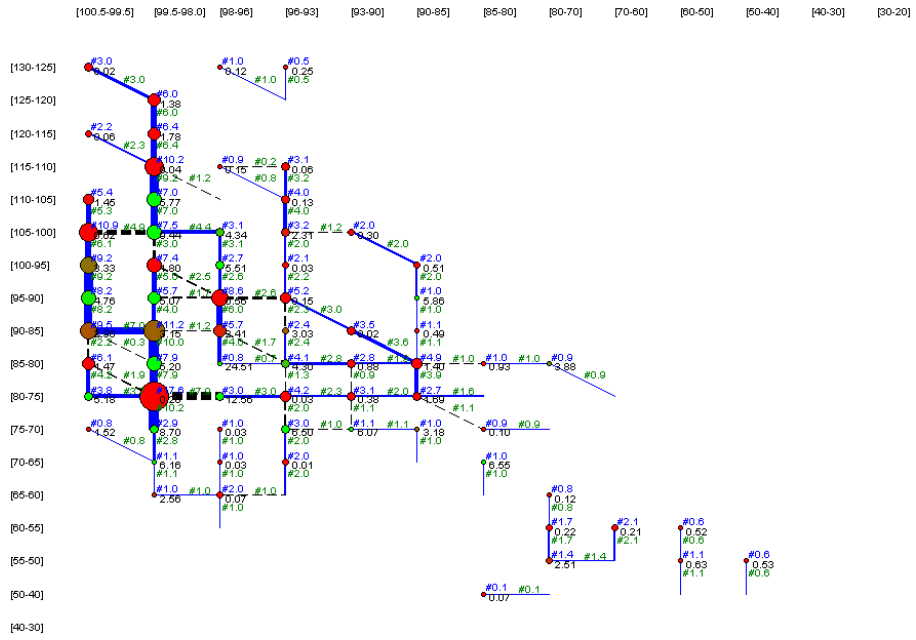
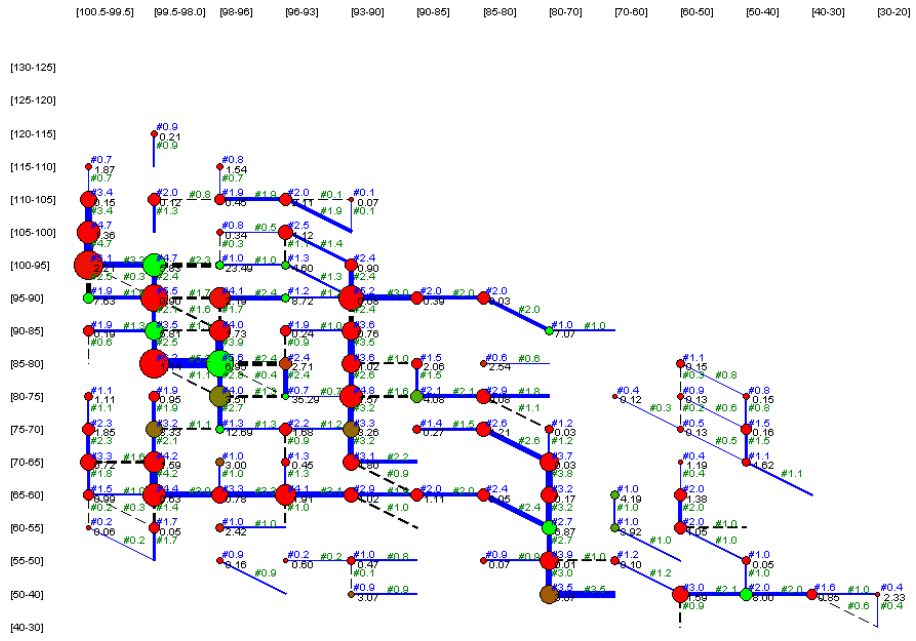


Figure 35: Visualization of the state transition trends and expected state and edge visiting count for all patients. Visualization scheme: the strongest transition among the three instantaneous links from each state are shown in blue while other transitions are drawn in dotted black. The line width and the node size reflect the expected count. The node color represents the average sojourn time (red to green: 0 to 5 years and above).

transition among the three instantaneous links are shown in blue. The node size reflects the relative number of eyes, the node color reflects the average dwelling time (red to green: 0 year to 5 years or above), the width of the link between nodes reflect the number of observed progression along the path. Several predominant paths were detected: early disease phase with RNFL thinning without an apparent change in VFI (thick vertical paths), and at RNFL range [8085], the transition trend reverses and VFI changes become more evident (blue horizontal paths). This *L shape* in progression supports the finding in [95] that RNFL thickness of around 77 microns acts as a tipping point at which functional deterioration became clinically observable with structural deterioration. We can identify several L shape dominant state transition



(a) age group (≤ 70)



(b) age group (> 70)

Figure 36: Visualization of the state transition trends and expected state and edge visiting count for age group (≤ 70) and (> 70). The visualization scheme is the same as in Fig. 35.

in this figure (at (VFI:100, RNFL [85 80]), (VFI:[98-96], RNFL:[80 75]), (VFI:[96-93], RNFL:(95 90)), etc.). At these corner point of L shapes, the model reveals that

the functional degeneration becomes more likely than the structural changes. Our 2D CT-HMM model efficiently visualize the non-linear relationship between structural and functional degeneration, which helps understanding glaucoma progression comprehensively.

In Fig. 36, we show the models trained by visits separated from two different age ranges (≤ 70) and (> 70). From Fig. 36(a), we see that for younger age group, the dominant structural changes are more apparent (the left-most vertical line) when vision is still perfect, and when RNFL is at $[90-85]$ there is a visual field degeneration trend. In Fig. 36(b) for age > 70, the trend of visual field loss starts when RNFL is at $[100-95]$, which is higher than the younger group, and at the successive RNFL ranges $[95-90]$ and $[90-85]$ there are also trends of visual field loss. This reveals that for older ages, visual field is susceptible to change even when RNFL is not very thin, while in younger ages, the visual function starts to show deficits until RNFL thickness is at lower ranges.

Unlike the commonly used methods for longitudinal analysis, namely trend or event analysis, the method presented here provides a comprehensive visualization of non-linear and complex nature of structural and functional glaucoma progression interactions. This facilitates accurate mapping of the structure and function relationship, decoding of clinical paths along the full spectrum of glaucoma severity, and prediction of the future progression.

5.2 Prediction of future states and measurements

Our approach to trajectory prediction is based on predicting the future state with the maximum probability, and then optionally predicting the continuous observation data for that state. First, the Viterbi decoding algorithm is used to decode the hidden state path for a subject based on past consecutive visits. Then the next state is given by $j = max_j P_{ij}(t)$, where P denotes the learned probability transition matrix from

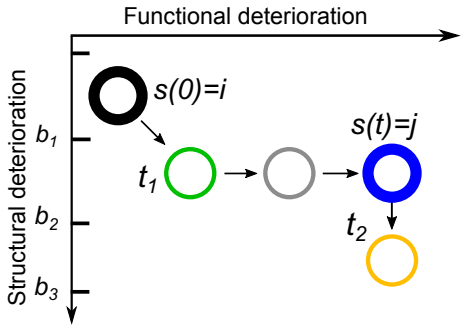


Figure 37: Illustration of the procedure of predicting continuous measurements at a future work for each data dimension based on transition probability matrix. Given the current state i at time 0, and a predicted future state at time t, then for interpolating the continuous value of structural marker at time t, we find time t_1 when the underlying process just enters a state with the same data range as state j, and time t_2 when the process just leaves this data range to the next level. These two time points can be found by binary search using $P(t)$ matrix.

a MD CT-HMM, i denotes the decoded state at the last past visit and t denotes the time interval between the last past visit and the next future visit.

Then, in order to predict the continuous measurements for each data dimension, we must interpolate between two predicted states with adjacent data range, since each state is associated with a range of measurements rather than a single measurement. For each data dimension, we identify the time t_1 when the patient first enters a state with the same data range as the predicted state j, and the time interval t_2 when the patient leaves the predicted state j and enters a state with the next adjacent data range. Both t_1 and t_2 can be found by analyzing the transition probability matrix. The measurement at time t can then be computed by linear interpolation between t_1 and t_2 and their corresponding measurements. This process can be repeated to predict subsequent states and associated measures.

5.2.1 Comparison to Bayesian Joint Linear Regression method

The main assumption of using linear regression method for prediction is that the current rates of disease progression will remain relatively unchanged unless further

interventions are delivered and, thus could be good predictors for future observations. In Bayesian joint linear regression method [58], linear regression is assumed for each response process, but the processes of each response are associated by imposing a joint multivariate distribution on the regression parameters (two slopes and two intercepts in Glaucoma case, a total of four parameters [58]) as the prior for influencing the parameter estimation for each individual eye. The parameters of this multivariate distribution as prior are estimated from the training data.

The formulation we adopt is detailed as follows [92]: $Y_{k,i} = \alpha_k + \beta_k X_{k,i} + \epsilon_{k,i}$ where k represents the measurement type, i represents the visit index, $X_{k,i}$ is the time of the visit, $Y_{k,i}$ is the measurement, α is the intercept, β is the slope, and $\epsilon_{k,i}$ is the residual between the actual measurement $Y_{k,i}$ and the fitted value $(\alpha_k + \beta_k X_{k,i})$. Suppose that $\epsilon_{k,i}|X_{k,i} \sim N(0, \sigma_k^2)$, i.e., $Y_{k,i}|X_{k,i} \sim N(\alpha_k + \beta_k X_{k,i}, \sigma_k^2)$. The likelihood function is [92]

$$\prod_{i=1}^{n} f(X_i, Y_i) = \prod_{i=1}^{n} f(X_i) f(Y_i|X_i) = L_1 L_2.$$

The first term L_1 does not involve the parameters of interests, so we can skip it. The second term

$$L_2 = \prod_{k=1}^{K} \prod_{i=1}^{n} f(Y_{k,i}|X_{k,i}) \propto \sigma_k^{-n} exp\{\frac{-1}{2\sigma_k^2} \sum_{i=1}^{n} (Y_{k,i} - (\alpha_k + \beta_k X_{k,i}))^2\},$$

where K is the number of response variables, which are 2 in Glaucoma progression model for one structural and one functional response. For the prior term, we use a multivariate Gaussian distribution for $\{\alpha_k, \beta_k\}_{k=1,2}$, and an non-informative prior for the error standard deviation term $\{\sigma_k\}_{k=1,...,K}$. Then, for each individual eye, we can thus find the Maximum a Posterior (MAP) estimation of $\{\alpha_k, \beta_k, \sigma_k\}_{k=1,2}$ by maximizing the product of the likelihood term L_2 and the prior term $p(\alpha_k, \beta_k, k = 1, .., K)$ for the regression parameters and $p(\sigma_k, k = 1, ..., K)$ for the standard deviation of error.

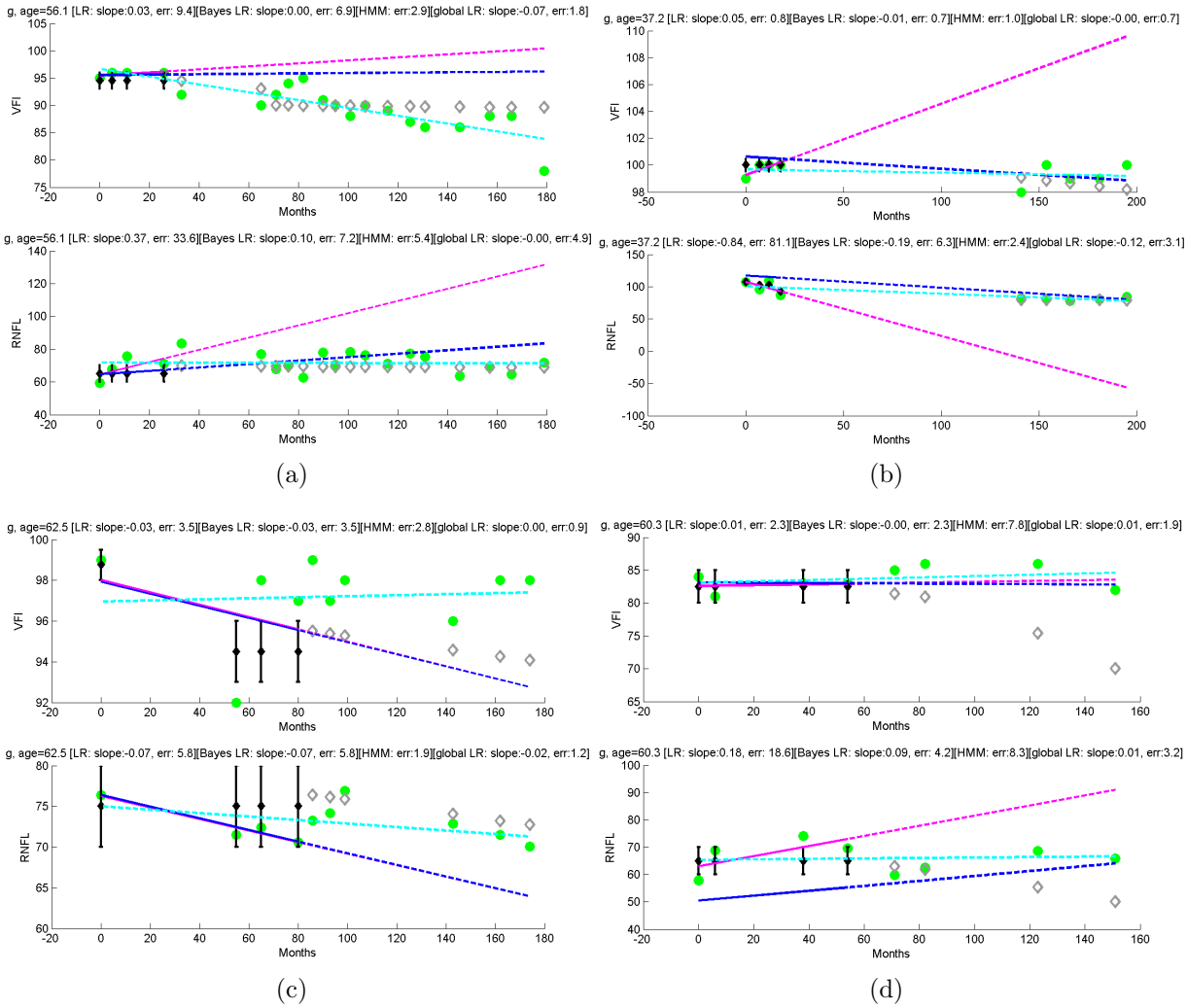


Figure 38: Example results of prediction (Magenta line: linear regression; blue line: Bayesian joint linear regression; light blue line: the ground truth global linear regression results which uses all data points; black range bar: CT-HMM decoded states for prior visits; gray diamond: CT-HMM predictive values. From example (a-c), we can see that CT-HMM gives reasonably good results; Bayesian method generally derives lower slope values than linear regression ans is more robust to noise. In example (d) we shows a failed example of CT-HMM, which overestimate the progression for VFI due to overfitting in a state where the training data is scarce.

5.2.2 Experimental results on prediction

We have conducted an initial study to assess the prediction ability of a 2D CT-HMM model using VFI and mean RNFL thickness for glaucoma progression (unpublished data). 101 glaucomatous eyes from 74 patients were followed for an average

Table 13: The mean absolute error (MAE) of the two measures (VFI, RNFL) using our 2D CT-HMM, Bayesian Joint Linear Regression (BJLR), and the conventional linear regression (LR) model. T-test results show that our method performs significantly better than both the competing models.

MAE	CT-HMM (H-Unif)	CT-HMM (S-Expm)	BJLR	LR
VFI	4.51 ± 10.38	4.64 ± 10.06	5.57 ± 11.11 *	7.00 ± 12.22 *
RNFL	7.09 ± 6.49	7.05 ± 6.57	9.65 ± 8.42 *	18.13 ± 20.70 *

of 11.7±4.5 years, and each eye has at least 5 visits (average 7.1±3.1 visits). 10-fold cross validation was used to evaluate the prediction performance. Testing proceeded by decoding the first 4 visits using a learned CT-HMM model and then predicting future observations. The mean absolute error (MAE) between the predicted values and the actual measurements was used for performance assessment. The performance of our model was compared to both conventional linear regression and Bayesian Joint Linear Regression. For linear regression, the individualized slope was computed from the first 4 prior visits, while for Bayesian method, the best intercepts and slopes were derived using both the four prior visits and the prior distribution of the parameters computed from the training set.

Our experimental results are shown in Table 13. Our results show that our 2D CT-HMM performs significantly better than both conventional linear regression model and the state-of-the-art Bayesian joint linear regression model. In Fig. 38, we show several examples of prediction results. From example (a-c), we can see that CT-HMM gives reasonably good results, and Bayesian method generally derives lower slope values than linear regression and Bayesian is more robust to noise or natural fluctuation. In example (d) we shows a failed example of CT-HMM, which overestimate the progression for VFI due to overfitting in a state where the training data is scarce. This error can be mitigated when the dataset is enlarged to include more patients in severe states. Otherwise, one may resort to using a hybrid approach, for example, using the average of Bayesian and CT-HMM results when the predictive future path is passing

states where the training data is insufficient.

5.3 Conclusion and future work

A novel state-based longitudinal analysis method based on multi-dimensional CT-HMMs is proposed for modeling the asymmetric and asynchronous aspects between structural and functional changes in glaucoma. The state-based continuous time modeling links the discrete irregularly-spaced measurements to an underlying hidden continuous-time process which indexes a progression in combinatorial disease states. Its flexible state-based structure enables it to handle intricate patterns between structural and functional changes without having restrictions on the global behavior. Our visualization results confirm the existence of a tipping point where the relationship between structural (RNFL thickness) and functional measures (visual field measurements) changes [95] with more prominent functional loss.

For predicting future continuous measurements, we devise a simple procedure to interpolate measures by inferencing on the transition probability matrix. Our prediction results significantly outperformed both the conventional linear regression-based methods and the start-of-the-art Bayesian joint linear regression method [62] for glaucoma prediction. The current clinical glaucoma management depends upon structural and visual function measurements with pre-specified observational intervals (typically every 6 or 12 months). With accurate state-based progression prediction, treatment plan can then be individualized with optimized visit intervals and testing schedules.

In sum, our promising results using M-D CT-HMM provides a useful starting point for developing a practical tool for glaucoma that can identify previously undiscovered progression phenotypes and may provide accurate glaucoma progression predictions.

For the future work, we propose the following two interesting directions:

M-D marker analysis: though RNFL thickness and the visual field parameters

VFI are the most used key markers for assessing glaucoma progression, there are many more clinical measures to assess glaucomatous damage other than these two. Optic disc parameters (e.g. cup/disc ratio, rim area, and cup volume) are the major ones among them. [10] [19] [74] [77] [76] [82]. They are also known to often disagree with RNFL thickness and visual field parameters [15] [12] [8] [47] [54].

In order to get comprehensive understanding about such complicated relationships, one can do pick any 2 or 3 biomarkers and analyze their relationship in the M-D state space. The resulting M-D CT-HMMs will reveal an array of non-linear relationships and interactions, many of them are currently undiscovered, among all the included clinical parameters. This will provide comprehensive understanding of the very complicated phenomenon called glaucoma progression.

Phenotype identification: National Eye Institute (NEI) defines glaucoma as a *group* of diseases that can damage the optic nerve and result in vision loss and blindness. In general, this group of diseases exhibits a similar path of vision deterioration, especially in the progression of visual field loss. However, as described above, there are many discordant reports on glaucoma progression. Therefore, it has become an accepted concept among glaucoma specialists that there potentially exist multiple glaucoma progression phenotypes, where a phenotype is a group of patients who are identified as having similar characteristics of progression. The potential phenotypes has never been scientifically and systematically identified yet.

As described above, Wollstein et al. reported a tipping point that divides the structure-function relationship into two fits.[95]. One line fit represents a group of eyes that have relatively thick retinal nerve fiber layer (RNFL) associated with near healthy visual field (VF), while the other fit shows a clear linear correlation between RNFL thickness and VF measurements. This tipping point suggests that there are 2 phases in glaucoma progression; (1) early or pre-perimetric glaucoma, and (2) once RNFL hits the tipping point both RNFL and VF deteriorate together. These two

phases can be viewed as two phenotypes. Pairing our analysis method with trajectory clustering techniques, potential phenotypes can be identified through both visualization by domain experts and by automated cluster analysis.

Overall, the conventional one size fits all approach to glaucoma progression modeling often treats minor phenotypes as noise, missing an opportunity to describe the atypical patterns that can improve treatments. With the guide of M-D CT-HMM analysis with trajectory clustering methods, the detection of glaucoma progression phenotypes will facilitate the classification of a given patient into one of the phenotypes, so that retrospective clinical information on the given prototype can be utilized for treatment planning.

CHAPTER VI

PRELIMINARY STUDY ON ALZEHIMER'S DISEASE AND HYPERTENSION USING MULTI-DIMENSIONAL CT-HMM

In this chapter, we present preliminary study on Alzehimer's disease and hypertension progression modeling using multi-dimensional CT-HMM. For Alzehimer's disease, we analyze and visualize the interaction of disease markers from three distinct aspects– structural, functional, and biochemical, over the course of disease evolution using a 3D CT-HMM. We will analyze the results and see whether they correspond to the current knowledge of the degeneration ordering. For hypertension, we analyze the two blood pressures, systolic and diastolic, from electronic health records (EHR) of a cohort population enrolled in a hypertension management program using a 2D CTHMM. We do cluster analysis on the decoded state transition trajectories from all patients. A novel visualization is presented to compare th states visited and transitions taken by two different subgroups with distinctive hypertension trajectories. Our model with visualization tools provides an interface for exploratory analysis and model validation, and improves the interpretability of the model results for healthcare researchers.

6.1 Applications on Alzheimer's disease progression modeling

Alzheimers Disease (AD) is an irreversible neuro-degenerative disease that results in a loss of mental function due to the deterioration of brain tissue. It is the most common cause of dementia among people over the age of 65, affecting an estimated 5.3 million Americans, yet no prevention methods or cures have been discovered. Recently, the

decreasing of amyloid beta in cerebral spinal fluid (CSF) has been found to be an early marker before detectable hippocampus atrophy. It will be beneficial to understand the relationship among clinical, imaging, biochemical biomarker characteristics as well as genetic factors on the entire spectrum of Alzheimers disease, as the pathology evolves, which may aid in preventing AD and developing treatments.

Our goal is to track the progression of the disease using available bio-markers to assess the brain's structural, bio-chemical, and functional changes over the course of disease evolution.

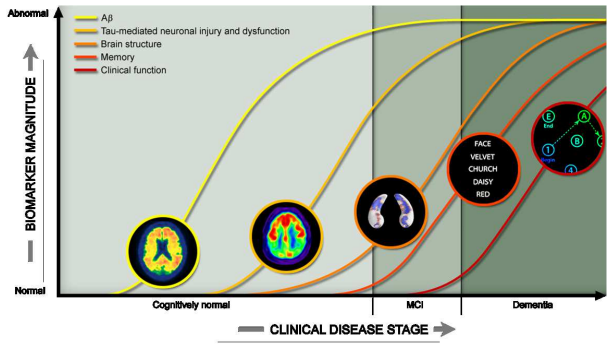


Figure 39: The current knowledge about the abnormality ordering of different biomarkers for Alzheimer's disease. The abnormality of amyloid beta level can be first detected (from CSF or from F18 amyloid imaging), then the tau level (from CSF or from PET scan), then the brain atrophy (MRI scan), then memory score, then cognition score (from ADNI website [84], used without permission).

6.1.1 3-D exploratory analysis of structural, functional, and biochemical markers

In this experiment, we analyze the temporal interaction among the three kinds of markers: amyloid beta ($A\beta$) level in cerebral spinal fluid (CSF) (bio-chemical marker), hippocampus volume (structural marker), and ADAS cognition score (functional marker) over the course of the disease. We obtained the *ADNI* (The *Alzheimers Disease Neuroimaging Initiative*)1 dataset from the website [84]. The mild cognition impairment (MCI) and AD patients who have at least two visits of all three indicated markers are included for our analysis, which results in 206 subjects of 2.38 ± 0.66 visits

traced in 1.56 ± 0.86 years with only 3 distinct time intervals in month resolution. A 3-D gridded state space with forwarding links is defined such that for each marker, we have 14 bands that span its value range. The procedure for constructing the state space and the definition of data emission model is the same as in the Glaucoma experiment. 277 states are instantiated and the model is then trained using *Soft(Expm)*. The running time using *Soft(Expm)* is about 17 minutes per iteration on a 2.67 GHz machine (for comparison, *Hard(Unif)* is around 2 minutes and *Eigen* fails.)

The 3D visualization result is shown in Fig. 40. The state transition trends show that the abnormality of $A\beta$ level emerges first (blue lines) when cognition scores are still normal. Hippocampus atrophy happens more often (green lines) when $A\beta$ levels are already low and cognition has started to shown abnormality. Most cognition degeneration happens (red lines) when both $A\beta$ levels and Hippocampus volume are already in abnormal stages. Our quantitative visualization results supports the recent findings that the decreasing of $A\beta$ level in CSF is an early marker before detectable hippocampus atrophy in cognition-normal elderly [17]. The CT-HMM disease model armed with 2D/3D interactive visualization functions can be utilized as an exploratory tool to gain insights of the disease progression and generate hypothesis to be further investigated by medical researchers.

In the future, we plan to integrate ADNI 1, 2, and GO data [84] so that more patients in very early disease stages can be included, and more informed disease progression picture can be derived.

6.1.2 Future directions

Below we list several future directions for further analyzing Alzheimer's disease:

- **3-D CT-HMM progression models with cluster analysis:** For each dimension, such as structural, functional, and biochemical, we may also first summarize the measurements to a few abstract states using clustering techniques,

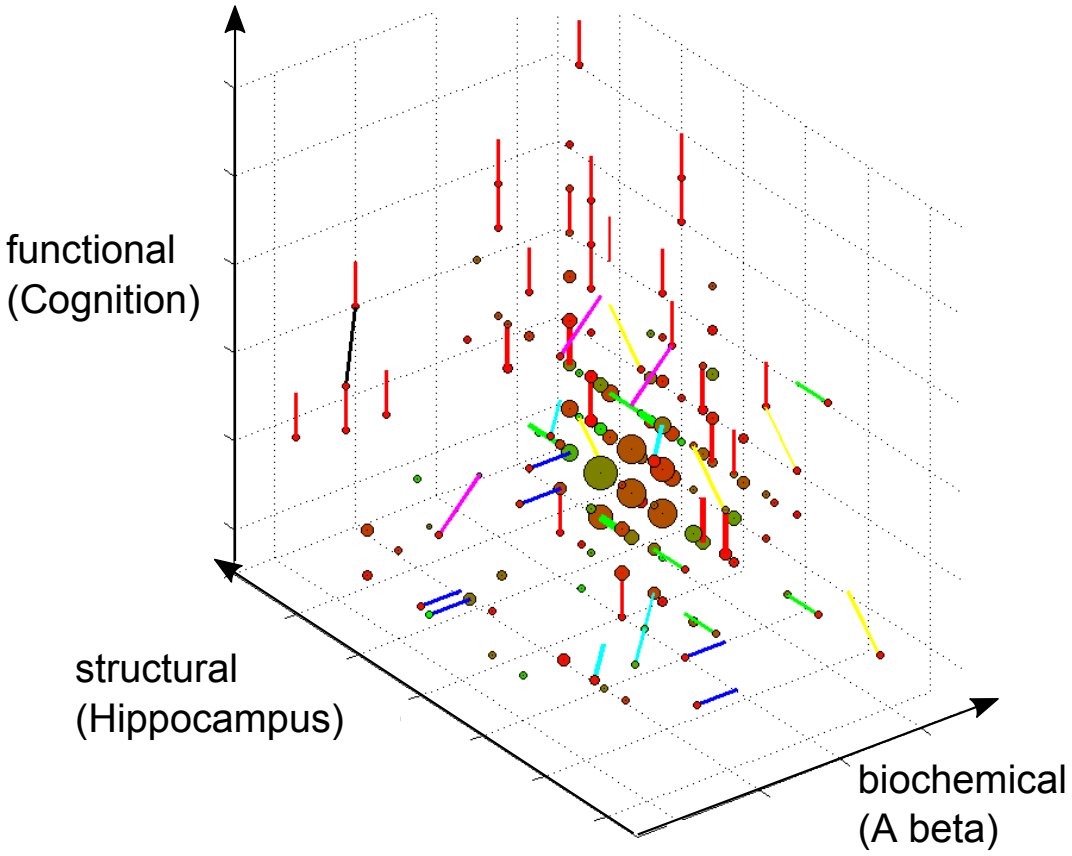


Figure 40: 3D CT-HMM analysis by using biochemical measures (amyloid beta), structural measures (hippocampus volume), and functional measure (ADAS cognition score), using ADNI1 dataset [84]. The visualization scheme: the strongest transition link from each state is color coded as follows: $A\beta$ direction only (blue), *hippo* only (green), *cog* only (red), $A\beta + hippo$ (cyan), $A\beta + cog$ (magenta), $hippo + cog$ (yellow), $A\beta + hippo + cog(black)$. The line width and the node size reflect the expected count. The node color represents the average sojourn time (red to green: 0 to 3 years and above).

and then use these states to define a single dimension. To achieve this, we may first train a left-to-right HMM using associated measures, and using the learned mean and covariate to represent the state property in that dimension. This way we can have a 3-D progression model which is easier to comprehend but much more measurement types are incorporated in one model.

- **CT-HMM with complete graph for event ordering analysis** As a prior work shown in Section 2.3.3, it is of interest to find the atrophy event ordering of each anatomical area for Alzheimer's disease. To this end, one idea is as follows. We can first find clusters from cross-sectional structural measurements in order to define a set of dominate states. Then, in order to find the temporal ordering of these states and then derive event ordering, we can model the temporal interactions between these states by using CT-HMM with full pairwise connections.

- **Spatial-temporal CT-HMM for longitudinal image analysis** One idea is to construct a spatial-temporal CT-HMM for capturing the temporal and spatial interactions of multiple anatomical areas for Alzheimer's disease. Each state represents structural properties (e.g., the texture, intensity, thickness, volume) features, at some time point for one anatomical area. The spatial links models the spatial relationship (co-occurrence) between anatomical areas. The temporal links are used to model the temporal structural changes for each area. By learning the model from a set of manually-segmented longitudinal images, the CT-HMM captures the dynamic structural changes of each area. This model can be used for understanding disease evolution directly in image level and can also be utilized as a time-varying prior for longitudinal image processing tasks, such as denoising and segmentation.

6.2 Applications on Hypertension progression modeling using Electronic Health Data (EHR) data

Hypertension is one of the main risk factors for cardiovascular disease and stroke, which claim over 17 million lives every year [99]. Several classes of drugs have been developed to help hypertensive patients maintain normal blood pressure, although the precise medication regimen needed to achieve blood pressure (BP) control for each individual is often difficult to determine. As a result, the medication regimen is often modified on a trial-and-error basis. Long term hypertension management programs [63] [30] can help participants achieve and maintain controlled blood pressure by encouraging regular home monitoring, facilitating medication adjustments and tracking disease parameters over time in electronic medical records.

Recently, there have been attempts to model hypertension from such electronic health record (EHR) data. One retrospective study [35] found that including EHR data improved the performance of regression models and gradient tree boosting for predicting cardiovascular-related death within 5 years, when compared to models trained with only traditional risk predictors. Several other studies were also performed to identify measures that act as risk predictors for the development of hypertension [53] [9]. However, for the most part *these earlier works did not propose any framework for modeling the longitudinal progression of hypertension in diagnosed individuals.*

More recent studies have begun to focus on the prediction of significant events within hypertension progression. Sun et al. used aggregated EHR data to predict transitions between controlled and uncontrolled hypertension, and also to identify which features produced the best prediction performance [80]. This approach, however, requires constant monitoring of patients hypertension control status in order to determine the transition at the next clinical encounter. This highlights some of the difficulties of developing predictive models for hypertension given irregularly timed patient visits spanning only some segment of the disease progression.

In order to overcome some of these issues, our work models hypertension as a hidden continuous-time Markov process evolving on a discrete state space, where each state is characterized by a certain range of systolic and diastolic blood pressures. This model structure can represent the continuous evolution of disease states using measurement data that arrives irregularly in time, and can be used to fill in missing measurements and predict future trajectories of hypertension evolution. This class of model is referred to as a continuous time hidden Markov model (CT-HMM) [29] [6] [52]. It can also be used to study the specific effects of different medication regimens or individual-level features on hypertension evolution characteristics.

A basic challenge in using data-driven methods to improve healthcare is the development of appropriate visualization methods (viz). Viz enables health researchers to understand and validate a given model and perform exploratory data analysis, including the generation of hypotheses regarding disease progression. Without viz there is a danger that models that are fit to complex high dimensional datasets become black box analysis tools whose outputs cannot be explained or interpreted by health researchers.

Here, we presents FluxMap, the first interactive viz system for displaying and analyzing state-based disease progressions models based on the CT-HMM framework. FluxMap is designed to visualize state and transition properties of the CT-HMM and supports the identification and tracking of sub-groups of patients over time. There have been a few previous related works: Leiva-Murillo et al. used CT-HMM to model comorbidity of various diseases over time, illustrating their results using a static diagram of states and transitions [42]. However, only a small proportion of states and only the most dominant transitions can be included to avoid visual clutter. Another group used EventFlow software to look for common patterns in hypertensive drug usage and relate them to patient outcomes [55] [69]. FluxMap has the potential for use with any longitudinal state-based progression model, though here we present its

use in conjunction with a CT-HMM model of hypertension built from longitudinal EHR data. To the best of our knowledge, this work is the first to demonstrate the application of an interactive visualization tool to state-based hypertension progression modeling.

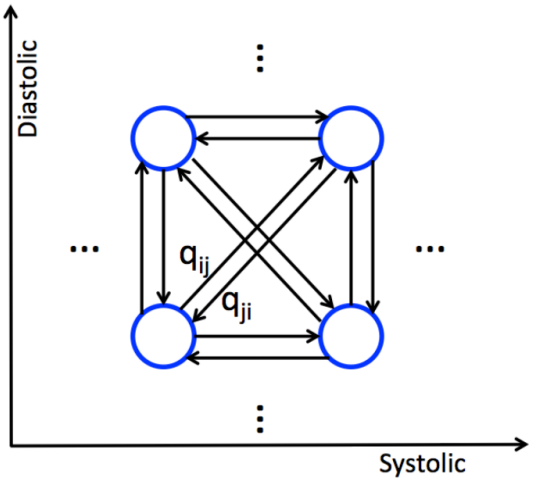


Figure 41: The state and link structure for our hypertension progression model. The links represent allowed instantaneous transitions with transition intensity q_{ij}

6.2.1 2-D exploratory analysis on blood pressure markers

We analyze EHR data from a patient cohort involved in a hypertension management program. The dataset provides longitudinal data from several patients, each having a variable number of irregularly timed visits. At each visit, a number of observed variables are measured, though these measurements may be incomplete in the sense that not every feature is measured at every time point; for example, lab tests are not done at every visit. Our hypertension model is based on a 2-dimensional discrete state space characterized by systolic and diastolic blood pressure range bands. Specifically, the discrete states are obtained by dividing the continuous space of each dimension into range bands. This discretization fits well with the typical practice of categorizing blood pressure using ranges of values. Each state is allowed to instantaneously transition to its adjacent states, and a q_{ij} parameter is to be estimated between each pair

of source and target state. An illustrative diagram is shown in Fig. 41. Our state-based model allows flexible transitions between states without assumptions such as linear or polynomial changes usually used in regression model, and thus is suitable for exploratory data analysis.

6.2.2 Interactive visualization

In order to convey the results of the CT-HMM model in an easily interpretable way, we opted for an interactive web-based visualization scheme. The visualization is implemented in JavaScript with the support of D3.js and jQuery libraries. The states are arranged on a 2D grid layout according to the states systolic and diastolic blood pressure ranges. Each state is represented as a circular node whose radius is proportional to the number of visits to that state. Transitions between states are bidirectional, so it was necessary to adopt an edge representation that encodes the direction of flow between the origin and destination nodes. This is important for being able to identify key behaviors in BP control like cycles or stable states.

We used asymmetric Bezier curves to depict transitions between states, where the straight section of the curve is oriented towards the origin node and the more curved section points towards the destination node. Again, the width and opacity of each edge are proportional to the number of transitions that occur in the subject population, to emphasize high-magnitude transitions. This weighting scheme was chosen over the scaling of edges based on transition intensities for several reasons. Using transition intensities could cause there to be links between every pair of states, however small, due to non-zero transition probabilities. The current weighting choice shows only the trajectories followed by subjects.

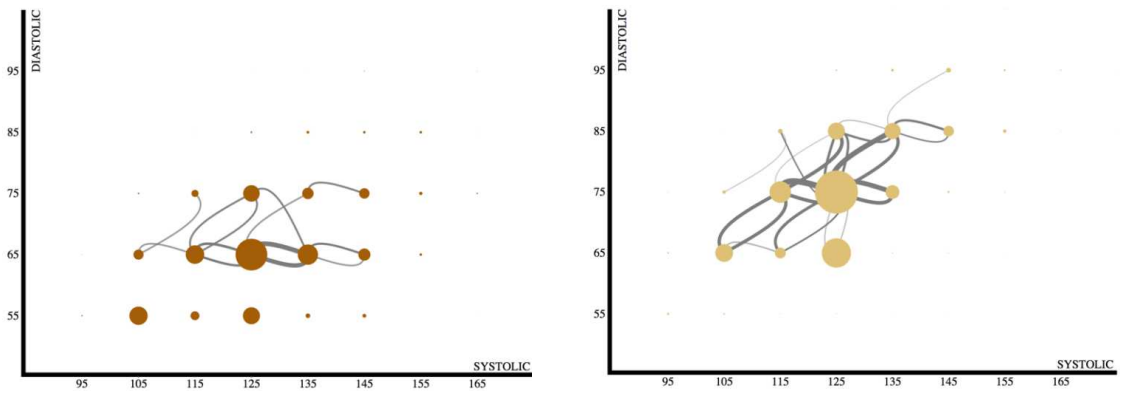
The second key component of the visual scheme is its interactivity. First, the user is allowed to interactively change the subject analysis population through filtering based on various features, such as medication class, BMI, risk level, lab test results,

etc. Besides using these standard filters, the subset population may also be chosen based on some complementary analysis, such as clustering of subjects based on similar trajectories; such a use case will be discussed shortly. Once a subset of the patient population is chosen, their collective behavior over the state space can be inspected to identify major visited states and transitions. Clicking on a particular node highlights incoming and/or outgoing links with other states by fading out the other nodes and links slightly. If multiple patient subgroups are selected, the states are depicted as pie graphs to depict the proportion of measurements from each subgroup. This design is appropriate for use with a small number of patient subgroups as a means of identifying any states visited predominantly by one subgroup.

This tool thus enables users to examine the disease progression model in the context of a particular set of features of interest. For example, a user may wish to compare the effects of incorporating different drug classes into a hypertension treatment regimen on the amount of time spent in controlled-BP states versus uncontrolled-BP states. To do this, the user simply selects only subjects taking the medications of interest to be represented in the graph, displaying the states visited and the proportions of each medication class at each one. Hence this visualization scheme combines an intuitive presentation of the CT-HMM model predictive results with the capacity for some visual exploratory analysis.

6.2.3 Trajectory clustering results

Trajectory clustering method: to demonstrate the exploratory visual analysis enabled by our tool, we first trained the CT-HMM on the dataset and produced decoded state sequences for all subjects for the window of time they were observed. The resulting sequences of states were compared between subjects to produce a similarity score based on the weighted Hamming distance. More specifically, the distance between two states is taken as the number of jumps or edits needed to move from



(a) Cluster 1 (b) Cluster 2

Figure 42: States and transitions by subjects in cluster 1 and 2 (From [85], used without permission).

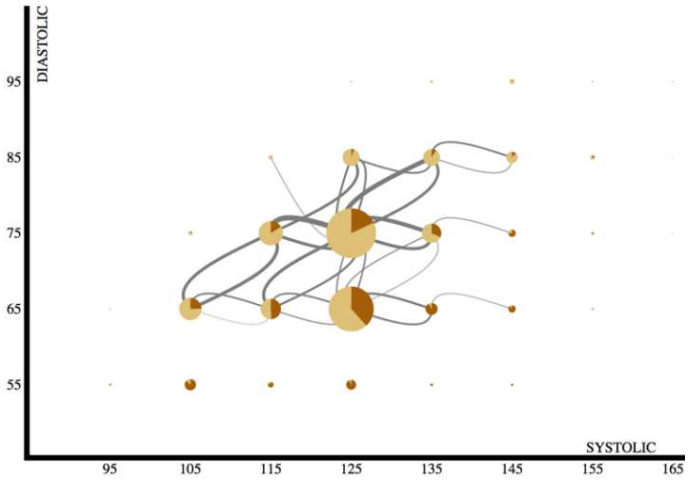


Figure 43: Clusters 1 and 2 shown together.

one state to the other. To compare two sequences, first an alignment is chosen such that the sequences overlap by at least 3 states; then, the cumulative distance between overlapping states is found and normalized. Different alignments are tested by sliding one sequence along the other, and the final similarity score is chosen as the minimum average distance over all possible alignments. This approach was adopted to account for the fact that subjects participation in the hypertension management program did not coincide with any distinctive time point in the progression of their hypertension. Finally, the subjects trajectories are grouped via hierarchical clustering to obtain

subsets with similar state sequences.

Clustering result: the above process applied on our dataset produced a hierarchical clustering of subjects, with two distinct low-level clusters depicted in Figures 42. Cluster 1 contains 194 subjects with lower diastolic BP on average compared to the 282 subjects in cluster 2. Further, it appears that subjects in cluster 1 often show changes in systolic pressure without accompanying changes in diastolic pressure, as evidenced by the slightly larger weights and numbers of horizontal transitions compared to vertical or diagonal ones. In contrast, subjects in cluster 2 have a noticeably higher tendency for coupled changes in systolic and diastolic BP, as shown by the prominent diagonal transitions in Figure 42.

These differences are evident in the combined view using pie charts shown in Figure 43. By limiting the number of classes to just 2, it is easy to see from the mostly dark brown nodes with diastolic BP 55 that these states are nearly exclusively visited by subjects in cluster 1. Similarly, states with diastolic BP 85 and higher are frequented by subjects in cluster 2 more than cluster 1. It is of interest to further analyze the characteristics of these two patient groups that may lead to the differences of BP change dynamics, such as age, occupations, medications, or existence of other diseases.

6.2.4 Future directions

We have presented an interactive visualization system for disease progression modeling using CT-HMM, with the potential for use as a validation tool for the model as well as an interface for conducting exploratory analysis.

The current visualization scheme has several possibilities for further development. At present, the identification of true transition sequences (with 2 or more transitions) is not possible due to the aggregation of transitions across subjects. While the current approach provides a summary overview of major transitions, it is necessary to include

some mechanism for recognizing and choosing specific sequences of states that are visited as part of a major path or cycle. These capabilities will become possible as the visualization scheme matures. Another possible limitation of the FluxMap visualization is its limitation to a 2-dimensional state space, whereas the CT-HMM model for which it was developed can support a higher-dimensional state space. This disparity can be partly addressed by adopting a side-by-side paneled view, in which each panel shows a 2D projection or collapse of the ND state space. However, this drastically reduces the interpretability of the visualization scheme and is untenable for more than 4 dimensions.

6.3 Conclusion

In this chapter, we use MD CT-HMM as an exploratory tool to analyze and visualize the state transition trends using the key markers from Alzheimer's disease and hypertension. For Alzheimer's disease, our quantitative visualization results using a 3D CT-HMM supports the recent findings that the decreasing of $A\beta$ level in celebral spinal fluid (CSF) is an early marker before detectable hippocampus atrophy in cognition-normal elderly [17], as the state transition trends show that the abnormality of $A\beta$ level emerges first when cognition scores are still normal, and Hippocampus atrophy happens more often when $A\beta$ levels are already low and cognition has started to shown abnormality.

For hypertension, a 2-D CT-HMM is trained to analyze the dynamic interactions between two blood pressures: systolic and diastolic. We also present a 2-D interactive visualization system for disease progression modeling using CT-HMM, with the potential for use as a validation tool for the model as well as an interface for conducting exploratory analysis. We demonstrated one use case of the tool for viewing states and trajectories; the case compared subjects from two different subgroups of hypertensive patients obtained by a hierarchical clustering of state sequences decoded by

the CT-HMM. This visualization framework can be extended to examine more elaborate questions related to the effects of different features on the trajectories taken by subjects. This may be useful for identifying phenotypes or different behaviors within hypertension management.

In sum, our results show that CT-HMM disease model armed with 2D/3D interactive visualization functions can be utilized as an exploratory tool to gain insights of the disease progression and generate hypothesis to be further investigated by medical researchers.

CHAPTER VII

CONCLUSION AND FUTURE WORK

In this dissertation, we develop multi-dimensional continuous-time hidden Markov models (M-D CT-HMM) to model disease progression. The continuous-time model is more suitable for irregularly-sampled temporal data such as clinical measurements than the discrete-time model. The M-D gridded state structure captures the co-evolution of multiple disease markers over time, and the learned dynamics among states can be visualized in a full spectrum of disease progression. Our M-D models can act as a general exploratory tool which bridges the gap between the simple 1-D disease staging models used often in clinics and the complex disease-specific models in research. The findings from using our tools can also be utilized to improve the design of disease-specific model. We also conduct conceptual comparison of our model to several existing state-based disease models under different disease contexts and applications. We find that our model can complement these existing models in flexibly revealing multi-factor interactions.

To learn large-scale CT-HMM efficiently, we develop novel EM learning algorithms leveraging the recent findings for CTMC [25] in evaluating the end-state conditioned expectations. To the best of our knowledge, we present the first comprehensive framework for parameter learning in CT-HMM, which both extends and unifies prior work on CTMC models. Based on the finite number of observations, our framework discretizes the estimation of posterior state probabilities into a time-inhomogeneous hidden Markov model, and incorporates hard or soft approaches to estimating the hidden state distributions corresponding to the observations. The benefits and drawbacks of hard and soft decoding approaches combined with the three different methods (*Expm,*

Unif, Eigen) for computing the conditional expectations are analyzed and validated experimentally. Our results show that soft EM has higher accuracy than hard while hard EM can save computation time. *Expm* works well for soft EM approaches while *Unif* method may be the best choice in hard EM due to its decomposability in computing only the expectations specified by the two decoded end-states.

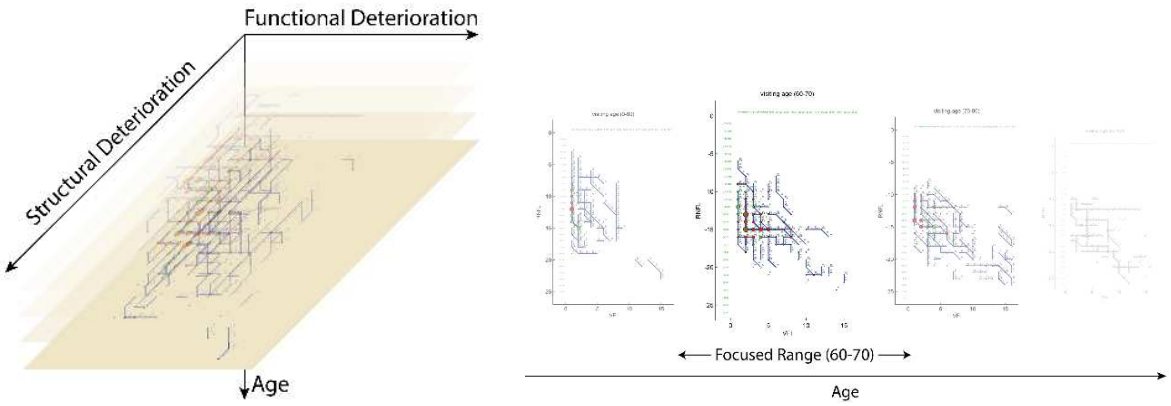
We also review the literature in finding the optimal state sequence marginaling out state dwell time given two end-states and a total time. We augment the state-of-the-art method in also computing the expected state durations for the optimal state path. We extend the three expectation evaluation methods (*Expm, Unif, Eigen*) in conjunction with a new close-form in computing the path-and-time conditioned expected state duration with simulation results. These decoding methods can be utilized in understanding the optimal hidden transition behaviors of a disease or decoding hidden trajectory of patient data.

We demonstrate the use of M-D CT-HMMs to three disease contexts, Glaucoma, Alzheimer's disease, and Hypertension. For Glaucoma, our results using a 2-D CT-HMM supports the finding in [95] that retinal nerve fiber thickness of around 77 microns acts as a tipping point at which functional deterioration became clinically observable with structural deterioration. In addition, our 2-D CT-HMM equipped with a simple procedure to interpolate future continuous measurements gives better prediction results than the state-of-the art method using Bayesian joint linear regression [62]. This results show the potential value of M-D CT-HMM for cost-efficient disease management in clinics. For Alzheimer's disease, our visualization results of the learned 3-D CT-HMM support the recent findings that the decreasing of $A\beta$ level in cerebral spinal fluid is an early marker before detectable hippocampus atrophy in cognition-normal elderly [17]. For hypertension, a 2-D CT-HMM defined over the ranges of the two blood pressure types is used. Two distinct patient groups are discovered by using trajectory clustering and the different characteristics can be clearly

visualized. This demonstrates the use of our model with visualization tool for exploratory data analysis which can generate new hypotheses such as transition trends and potential phenotypes to be further investigated by the domain experts.

In sum, we present the first comprehensive framework for parameter learning in CT-HMM, which both extends and unifies prior work on CTMC models. Our M-D CT-HMM model paired with visualization tools help capture and comprehend the possibly complex co-evolution behavior of multiple biomarkers, which can be utilized for predicting future progression trajectories and supporting the identification of potential phenotypes. This general exploratory tool for disease progression modeling bridges the gap of conventional 1-D disease model and complex disease-specific model, which has practical value for both clinics and research, leading to cost-effective disease management and more insights into the underlying mechanism of a disease.

Several future directions include:



(a) Perspective 3D view (b) Projected view using C-mode slicing

Figure 44: Illustration of 3D perspective view and 2D projection view with data slicing. While it is easy to grasp an overall picture with a 3D perspective view (a), 2D projection view is more suited to detailed exploration of the statistics (b). Users can switch back and forth between these two views on the fly. The data slicing range (in this example, the age dimension) can be interactively adjusted.

Interactive visualization tool for M-D CT-HMM: We have developed several 2D and 3D visualization tools for M-D CT-HMM which give a global picture of the progression of a cohort. Dependent variables, such as the number of records in a

given state and its dwelling time, can be displayed by size and color. Subpopulations of patients can be identified (e.g. by a common state they visit) and tracked over time. The distribution of other attributes, such as age, can be visualized at each state (pie graph view). In order to visualize CT-HMM with greater than 3 dimensions, we can use 2D projection and range slicing (c-mode slicing). Fig. 44 illustrates *interactive selection* of age ranges to update the 2D progression plot in real-time. These tools can enable medical experts to explore a higher dimensional model and gain insights into progression within selected data ranges.

Progression phenotypes identification: In Glaucoma disease, existence of various glaucoma progression phenotypes is an accepted concept among glaucoma specialists but has never been scientifically and systematically identified yet. M-D CT-HMM can be utilized to systematically identify potential glaucoma progression phenotypes by analyzing the learned population-level transition statistics or by grouping patients of similar decoded trajectories. The glaucoma experts can examine the automatically identified phenotypes and analyze whether there are any distinct characteristics of these patient groups. Furthermore, these phenotypes can be utilized for trajectory-based progression prediction. By training separate CT-HMMs for each phenotype, we can then classify a given new patient into one phenotype based on their data history, and perform future trajectory prediction.

The basic idea is to cluster patients into groups based on the similarity of their progression data. One method from [100] for longitudinal clustering with HMM is to compute mutual-fitness of the two sequences using the two models trained for each sequence separately, and do spectral clustering on the resulting similarity matrix, which can handle differing sequence length and is shown to be robust to noise.

Incorporation of covariate effects in large-scale CT-HMM and estimation of model parameters simultaneously: Besides learning the characteristics of disease progression using the key disease markers, incorporating and studying the

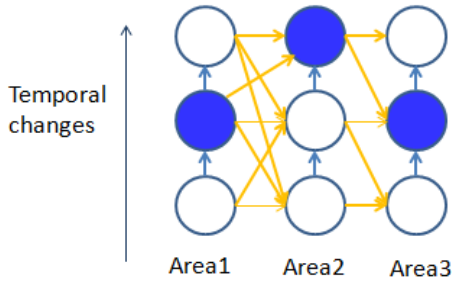


Figure 45: The spatial-temporal CT-HMM for longitudinal medical image analysis.

effects of covariates (such as age, race, commorbidities, treatments) in the state transition rate is also of great clinical interest, which may also result in higher predictive power. Incorporating the covariate effects to construct individualized transition rate matrix using cox proportional model is an established method. However, how to estimate the weight parameters for these covariates together with learning the baseline transition rates (when weights are set to zero) appears to be unsolved, and is an important future work.

M-D CT-HMM / spatial-temporal CT-HMM for longitudinal image analysis: CT-HMM can be useful for longitudinal medical image analysis. To our knowledge, it has not been applied to image analysis yet. One intuitive idea is to use M-D CT-HMMs as a dynamic prior for longitudinal image processing. The discrete states for structural features can be defined based on training a 1-D left-to-right model first. The M-D model can then be built for structural, functional, and other aspects. This model can predict the future state distribution given the current time point, which can be utilized as a prior in conducting longitudinal image processing.

Another related idea is to construct a spatial-temporal CT-HMM for capturing the temporal and spatial interactions of multiple anatomical areas (see Fig. 45). Each state represents structural properties (e.g., the texture, intensity, thickness, volume) features, at some time point for one anatomical area. The spatial links models the spatial relationship (co-occurrence) between anatomical areas. The temporal links are used to model the temporal structural changes for each area. By learning the model

from a set of manually-segmented longitudinal images, the CT-HMM captures the dynamic structural changes of each area. This model can be used for understanding disease evolution directly in image level and can also be utilized as a time-varying prior for longitudinal image processing tasks, such as denoising and segmentation.

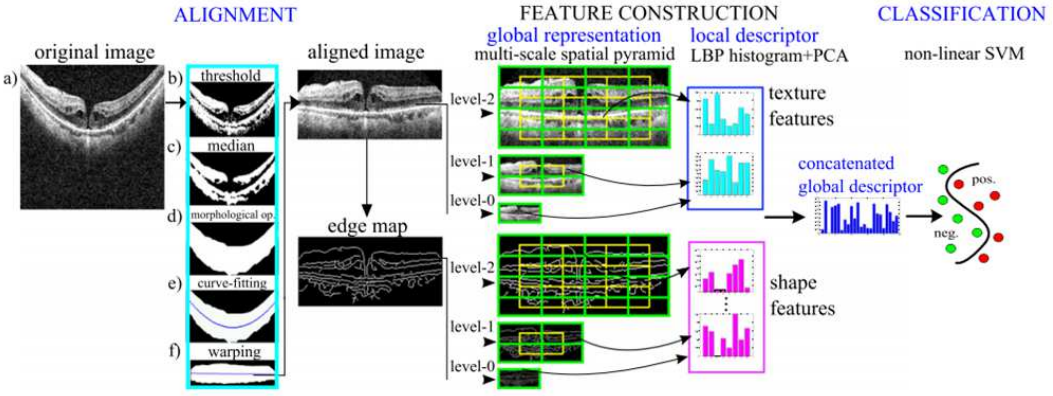


Figure 46: The framework for macular pathology classification in retinal OCT images from our prior work ([48],[49],[50]

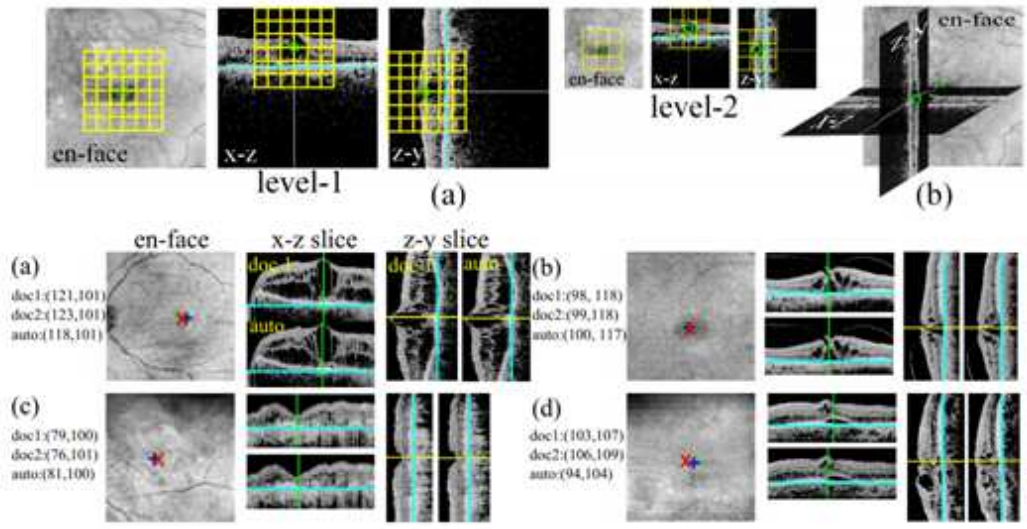


Figure 47: Foveola localization in retinal 3D OCT images using structural support vector machine from our prior work ([51])

We can apply this idea on modeling longitudinal structural changes in Glaucoma directly on Optical Coherence Tomography (OCT) images. In pursuing this future

direction, we can leverage our prior work for automated macular pathology classification in retinal OCT images ([48],[49],[50] see Fig. 46) and automated foveola localization in retinal OCT images ([51], see Fig. 47) to automatically do registration and generate observational features for longitudinal image analysis.

REFERENCES

[1] Alencar, L., Zangwill, L., Weinreb, R., and et al., "A comparison of rates of change in neuroretinal rim area and retinal nerve fiber layer thickness in progressive glaucoma," *Invest Ophthalmol Vis Sci*, vol. 51, no. 7, pp. 3531–9, 2010.

[2] Artes, P. and B.C., C., "Longitudinal changes in the visual field and optic disc in glaucoma," *Prog Retin Eye Res*, vol. 24, no. 3, pp. 333–54, 2005.

[3] Artes, P., Nicolela, M., LeBlanc, R., and et al., "Visual field progression in glaucoma: total versus pattern deviation analyses," *Invest Ophthalmol Vis Sci*, vol. 46, no. 12, pp. 4600–6, 2005.

[4] Asmussen, S., Olsson, M., and Nerman, O., "Fitting phase-type distributions via the EM algorithm," *Scand. J. Statist*, vol. 23, no. 4, pp. 419–441, 1996.

[5] Bandyopadhyay, S., Ganguli, B., and Chatterjee, A., "A review of multivariate longitudinal data analysis," *Statistical Methods in Medical Research*, vol. 20, pp. 299–330, 2011.

[6] Bartolomeo, N., Trerotoli, P., and Serio, G., "Progression of liver cirrhosis to HCC: an application of hidden markov model," *BMC Med Research Method.*, vol. 11, no. 38, 2011.

[7] Bladt, M. and Srensen, M., "Statistical inference for discretely observed markov jump processes," *J. R. Statist. Soc. B*, vol. 39, no. 3, p. 395410, 2005.

[8] Bowd, C., Lee, I., Goldbaum, M., and et al., "Predicting glaucomatous progression in glaucoma suspect eyes using relevance vector machine classifiers for combined structural and functional measurements," *Invest Ophthalmol Vis Sci*, vol. 53, no. 4, pp. 2382–9, 2012.

[9] Bozorgmanesh, M. and et al., "A point-score system superior to blood pressure measures alone for predicting incident hypertension: Tehran lipid and glucose study," *Journal of hypertension*, vol. 29, no. 8, pp. 1486–1493, 2011.

[10] Burgansky-Eliash, Z., Wollstein, G., Bilonick, R., and et al., "Glaucoma detection with the heidelberg retina tomograph 3," *Ophthalmology*, vol. 114, no. 3, pp. 466–71, 2007.

[11] Burgansky-Eliash, Z., Wollstein, G., Chu, T., and et al., "Optical coherence tomography machine learning classifiers for glaucoma detection: a

preliminary study," *Invest Ophthalmol Vis Sci*, vol. 46, no. 11, pp. 4147–52, 2005.

[12] Chauhan, B., McCormick, T., Nicolela, M., and et al., "Optic disc and visual field changes in a prospective longitudinal study of patients with glaucoma: comparison of scanning laser tomography with conventional perimetry and optic disc photography," *Arch Ophthalmol*, vol. 119, no. 10, pp. 1492–9, 2001.

[13] Chen, H. H., Duffy, S. W., and Tabar, L., "A markov chain method to estimate the tumour progression rate from preclinical to clinical phase, sensitivity and positive predictive value for mammography in breast cancer screening," *The Statistician*, vol. 45, no. 3, pp. 307–317, 1996.

[14] Chen, H. H., Duffy, S. W., and Tabar, L., "A markov chain method to estimate the tumour progression rate from preclinical to clinical phase, sensitivity and positive predictive value for mammography in breast cancer screening," *The Statistician*, vol. 45, no. 3, 1996.

[15] Cioffi, G., Liebmann, J., C.A., J., and et al., "Structural-functional relationships of the optic nerve in glaucoma," *J Glaucoma*, vol. 9, no. 1, pp. 3–4, 2000.

[16] Cox, D. R. and Miller, H. D., *The Theory of Stochastic Processes*. London: Chapman and Hall, 1965.

[17] Fagan, A. M., Head, D., Shah, A. R., and et. al, "Decreased CSF A beta 42 correlates with brain atrophy in cognitively normal elderly," *Ann Neurol.*, vol. 65, no. 2, p. 176183, 2009.

[18] Fonteijn, H. M. and et al., "An event-based disease progression model and its application to familial alzheimer's disease and huntington's disease," *NeuroImage*, vol. 60, no. 3, 2012.

[19] Gabriele, M., Wollstein, G., Bilonick, R., and et al., "Comparison of parameters from heidelberg retina tomographs 2 and 3," *Ophthalmology*, vol. 115, no. 4, pp. 673–7, 2008.

[20] Gatti, E., Luciani, D., and Stella, F., "A continuous time bayesian network model for cardiogenic heart failure," *Flexible Services and Manufacturing Journal*, vol. 23, p. 496515, 2012.

[21] Guedes, V., J.S., S., Hertzmark, E., and et al., "Optical coherence tomography measurement of macular and nerve fiber layer thickness in normal and glaucomatous human eyes," *Ophthalmology*, vol. 110, no. 1, pp. 177–89, 2003.

[22] Hajiaghayi, M., Kirkpatrick, B., Wang, L., and et al., "Efficient continuous-time markov chain estimation," in *Intl. Conference on Machine Learning*, 2014.

[23] Higham, N., *Functions of Matrices: Theory and Computation*. SIAM, 2008.

[24] Hobolth, A. and L.Jensen, J., "Statistical inference in evolutionary models of DNA sequences via the EM algorithm," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, 2005.

[25] Hobolth, A. and Jensen, J. L., "Summary statistics for endpoint-conditioned continuous-time markov chains," *Journal of Applied Probability*, vol. 48, no. 4, pp. 911–924, 2011.

[26] Jackson, C. H., L. D. Sharples, S. G. T., Duffy, S. W., and Couto, E., "Multistate markov models for disease progression with classification error," *Journal of the Royal Statistical Society, Series D - The Statistician*, vol. 52, no. 2, pp. 193–209, 2003.

[27] Jackson, C. H. and Sharples, L. D., "Hidden markov models for the onset and progression of bronchiolitis obliterans syndrome in lung transplant recipients," *Statistics in Medicine*, vol. 21, pp. 113–128, 2002.

[28] Jackson, C. H. and Sharples, L. D., "Hidden markov models for the onset and progression of bronchiolitis obliterans syndrome in lung transplant recipients," *Statistics in Medicine*, vol. 21, 2002.

[29] Jackson, C. H., "Multi-state models for panel data: the msm package for R," *Journal of Statistical Software*, vol. 38, no. 8, 2011.

[30] Jaffe, M. G. and et al., "Improved blood pressure control associated with a large-scale hypertension program," *JAMA*, vol. 310, no. 7, pp. 699–705, 2013.

[31] Jensen, A., "Markoff chains as an aid in the study of markoff processes," *Skand. Aktuarietidskr*, vol. 36, pp. 87–91, 1953.

[32] Kalbfleisch, J. D. and Lawless, J. F., "The analysis of panel data under a markov assumption," *Journal of the American Statistical Association*, vol. 80, no. 392, 1985.

[33] Kay, R., "A markov model for analysing cancer markers and disease states in survival studies," *Biometrics*, 1986.

[34] Kay, R., "A markov model for analysing cancer markers and disease states in survival studies," *Biometrics*, vol. 42, no. 4, pp. 855–865, 1986.

[35] Kennedy, E. H. and et al., "Improved cardiovascular risk prediction using nonparametric regression and electronic health record data," *Medical care*, vol. 51, no. 3, pp. 251–258, 2013.

[36] Kingman, S., "Glaucoma is second leading cause of blindness globally," *Bulletin of the World Health Organization*, vol. 82, no. 11, 2004.

[37] Kirby, A. J. and Spiegehalter, D. J., "Statistical modeling for the precursors of cervical cancer," *Case studies in Biometry*, 1994.

[38] Klotz, J. H. and Sharples, L. D., "Estimation for a markov heart transplant model," *The Statistician*, vol. 43, no. 3, pp. 431–436, 1994.

[39] Kotowski, J., Wollstein, G., Folio, L. S., Ishikawa, H., and Schuman, J. S., "Clinical use of OCT in assessing glaucoma progression," *Ophthal S L Imaging*, vol. 42, 2011.

[40] Kotowski, J., Wollstein, G., Folio, L. S., Ishikawa, H., and Schuman, J. S., "Clinical use of OCT in assessing glaucoma progression," *Ophthal S L Imaging*, vol. 42, 2011.

[41] Lange, J. M. and Minin, V. N., "Fitting and interpreting continuous-time latent markov models for panel data," *Statistics in Medicine*, vol. 32, no. 26, pp. 4581–95, 2013.

[42] Leiva-Murillo, J. M., Rodriguez, A. A., and Baca-Garcia, E., "Visualization and prediction of disease interactions with continuous-time hidden markov models," in *Advances in Neural Information Processing Systems*, 2011.

[43] Leung, C. K. S., Liu, S., Weinreb, R. N., Lai, G., Ye, C., Cheung, C. Y. L., Pang, C. P., Tse, K. K., and Lam, D. S. C., "Evaluation of retinal nerve fiber layer progression in glaucoma," *American Academy of Ophthalmology*, 2011.

[44] Leung, C., Cheung, C., Weinreb, R., and et al., "Evaluation of retinal nerve fiber layer progression in glaucoma: a study on optical coherence tomography guided progression analysis," *Invest Ophthalmol Vis Sci*, vol. 51, no. 1, pp. 217–22, 2010.

[45] Leung, C., Chiu, V., Weinreb, R., and et al., "Evaluation of retinal nerve fiber layer progression in glaucoma: a comparison between spectral-domain and time-domain optical coherence tomography," *Ophthalmology*, vol. 118, no. 8, pp. 1558–62, 2011.

[46] Levin, P., Lefebvre, J., and Perkins, T. J., "What do molecules do when we are not looking? state sequence analysis for stochastic chemical systems," *J. R. Soc. Interface*, vol. 9, no. 77, pp. 3411–25, 2012.

[47] Lin, D., Leung, C., Weinreb, R., and et al., "Longitudinal evaluation of optic disc measurement variability with optical coherence tomography and confocal scanning laser ophthalmoscopy," *J Glaucoma*, vol. 18, no. 2, pp. 101–6, 2009.

[48] Liu, Y.-Y., Ishikawa, H., Chen, M., Wollstein, G., Schuman, J. S., and Rehg, J. M., "Automated macular pathology diagnosis in retinal oct images using multi-scale spatial pyramid with local binary patterns," in *MICCAI*, 2010.

[49] Liu, Y.-Y., Ishikawa, H., Chen, M., Wollstein, G., Schuman, J. S., and Rehg, J. M., "Automated macular pathology diagnosis in retinal oct images using multi-scale spatial pyramid and local binary patterns in texture and shape encoding," *Medical Image Analysis*, 2011.

[50] Liu, Y.-Y., Ishikawa, H., Chen, M., Wollstein, G., Schuman, J. S., and Rehg, J. M., "Computerized macular pathology diagnosis in spectral domain optical coherence tomography scans based on multi-scale texture and shape features," *Investigative Ophthalmology and Visual Science*, 2011.

[51] Liu, Y.-Y., Ishikawa, H., Chen, M., Wollstein, G., Schuman, J. S., and Rehg, J. M., "Automated foveola localization in retinal 3d-oct images using structural support vector machine prediction," in *MICCAI*, 2012.

[52] Liu, Y., Ishikawa, H., Chen, M., and et al., "Longitudinal modeling of glaucoma progression using 2-dimensional continuous-time hidden markov model," *Med Image Comput Comput Assist Interv*, vol. 16, no. 2, pp. 444–51, 2013.

[53] Lukas, A. and et al., "Body mass index is the main risk factor for arterial hypertension in young subjects without major comorbidity," *European journal of clinical investigation*, vol. 33, no. 3, pp. 223–230, 2003.

[54] Mansouri, K., Leite, M., Medeiros, F., and et al., "Assessment of rates of structural change in glaucoma using imaging technologies," *Eye (Lond)*, vol. 25, no. 3, pp. 269–77, 2011.

[55] Margret, B. and Wu., S., "Workshop: Exploring temporal patterns in hypertensive drug therapy," *HCIL*, pp. 337–344, 2014.

[56] Marshall, G. and Jones, R. H., "Multi-state markov models and diabetic retinopathy," *Statistics in Medicine*, vol. 14, 1995.

[57] Marshall, G. and Jones, R. H., "Multi-state markov models and diabetic retinopathy," *Statistics in Medicine*, 1995.

[58] Medeiros, F. A., Zangwill, L. M., Girkin, G., Liebmann, J. M., and Weinreb, R. N., "Combining structural and functional measurements to improve estimates of rates of glaucomatous progression," *Am J Ophthalmol*, vol. 153, no. 6, 2012.

[59] Medeiros, F., Alencar, L., Zangwill, L., and et al., "Prediction of functional loss in glaucoma from progressive optic disc damage," *Arch Ophthalmol*, vol. 127, no. 10, pp. 1250–6, 2009.

[60] Medeiros, F., Lisboa, R., Weinreb, R., and et al., "Retinal ganglion cell count estimates associated with early development of visual field defects in glaucoma," *Ophthalmology*, vol. 120, no. 4, pp. 736–44, 2013.

[61] Medeiros, F., Zangwill, L., Alencar, L., and et al., "Rates of progressive retinal nerve fiber layer loss in glaucoma measured by scanning laser polarimetry," *Am J Ophthalmol*, vol. 149, no. 6, pp. 908–15, 2010.

[62] Medeiros, F., Zangwill, L., Girkin, C., and et al., "Combining structural and functional measurements to improve estimates of rates of glaucomatous progression," *Am J Ophthalmol*, vol. 153, no. 6, pp. 1197–205, 2012.

[63] Mehos, B. M., Saseen, J. J., and MacLaughlin, E. J., "Effect of pharmacist intervention and initiation of home blood pressure monitoring in patients with uncontrolled hypertension," *Pharmacotherapy: The Journal of Human Pharmacology and Drug Therapy*, vol. 20, no. 11, 2000.

[64] Metzner, P., Horenko, I., and Schtte, C., "Generator estimation of markov jump processes based on incomplete observations nonequidistant in time," *Physical Review E*, vol. 76, no. 066702, 2007.

[65] Minin, V. and Suchard, M., "Counting labeled transitions in continuous-time markov models of evolution," *Journal of Mathematical Biology*, vol. 56, no. 3, pp. 391–412, 2008.

[66] Perkins, T. J., "Maximum likelihood trajectories for continuous-time markov chains," *NIPS*, 2009.

[67] Quigley, H. A. and Vitale, S., "Models of open-angle glaucoma prevalence and incidence in the united states," *Invest Ophthalmol Vis Sci*, vol. 38, no. 1, 1997.

[68] Rabinar, L. R., "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, 1989.

[69] Rind, A. and et al., "Interactive information visualization for exploring and querying electronic health records: A systematic review," *HCIL-2010*, pp. 337–344, 2010.

[70] Ross, S. M., *Stochastic Processes*. New York: John Wiley, 1983.

[71] Russell, R., Malik, R., Chauhan, B., and et al., "Improved estimates of visual field progression using bayesian linear regression to integrate structural information in patients with ocular hypertension," *Invest Ophthalmol Vis Sci*, vol. 53, no. 6, pp. 2760–9, 2012.

[72] Satten, G. A. and Longini, I. M., "Markov chains with measurement error: Estimating the true course of a marker of the progression of human immunodeficiency virus disease," *Applied Statistics-Journal of the Royal Statistical Society Series C*, vol. 45, no. 3, 1996.

[73] Schell, G. J., Lavieri, M. S., Stein, J. D., and Musch, D. C., "Filtering data from the collaborative initial glaucoma treatment study for improved identification of glaucoma progression," *BMC Medical Informatics and Decision Making*, vol. 13, no. 137, 2013.

[74] Schuman, J., Wollstein, G., Farra, T., and et al., "Comparison of optic nerve head measurements obtained by optical coherence tomography and confocal scanning laser ophthalmoscopy," *Am J Ophthalmol*, vol. 135, no. 4, pp. 504–12, 2003.

[75] Sharples, L. D., "Use of the gibbs sampler to estimate transition rates between grades of coronary disease following cardiac transplantation," *Statistics in Medicine*, vol. 12, pp. 1155–1169, 1993.

[76] Strouthidis, N. and D.F., G.-H., "New developments in heidelberg retina tomograph for glaucoma," *Curr Opin Ophthalmol*, vol. 19, no. 2, pp. 141–8, 2008.

[77] Strouthidis, N., Gardiner, S., Owen, V., and et al., "Predicting progression to glaucoma in ocular hypertensive patients," *J Glaucoma*, vol. 19, no. 5, pp. 304–9, 2010.

[78] Strouthidis, N., Scott, A., Peter, N., and et al., "Optic disc and visual field progression in ocular hypertensive subjects: detection rates, specificity, and agreement," *Invest Ophthalmol Vis Sci*, vol. 47, no. 7, pp. 2904–10, 2006.

[79] Strouthidis, N., Vinciotti, V., A.J., T., and et al., "Structure and function in glaucoma: The relationship between a functional visual field map and an anatomic retinal map," *Invest Ophthalmol Vis Sci*, vol. 47, no. 12, pp. 5356–62, 2006.

[80] Sun, J. and et al., "Predicting changes in hypertension control using electronic health records from a chronic disease management program," *Journal of the American Medical Informatics Association*, vol. 21, no. 2, pp. 337–344, 2014.

[81] Tan, O., Chopra, V., A.T., L., and et al., "Detection of macular ganglion cell loss in glaucoma by fourier-domain optical coherence tomography," *Ophthalmology*, vol. 116, no. 12, pp. 2305–14, 2009.

[82] Tannenbaum, D., Zangwill, L., Bowd, C., and et al., "Relationship between visual field testing and scanning laser polarimetry in patients with a large cup-to-disk ratio," *Am J Ophthalmol*, vol. 132, no. 4, pp. 501–6, 2001.

[83] Tataru, P. and Hobolth, A., "Comparison of methods for calculating conditional expectations of sufficient statistics for continuous time markov chains," *BMC Bioinformatics*, vol. 12, no. 465, 2011.

[84] The Alzheimers Disease Neuroimaging Initiative, "ADNI website:http://adni.bmap.ucla.edu/,"

[85] The Eye Diseases Prevalence Research Group, "Causes and prevalence of visual impairment among adults in the united states," *Arch Ophthal*, no. 122, 2004.

[86] Titman, A. and Sharples, L., "Semi-markov models with phase-type sojourn distribution," *Biometrics*, vol. 66, no. 3, pp. 742–52, 2010.

[87] Townsend, K., Wollstein, G., Danks, D., and et al., "Heidelberg retina tomograph 3 machine learning classifiers for glaucoma detection," *Br J Ophthalmol*, vol. 92, no. 6, pp. 814–8, 2008.

[88] Uhry, Z. and et al., "Multi-state markov models in cancer screening evaluation: a brief review and case study," *Statistical Methods in Medical Research*, vol. 19, 2010.

[89] Van Loan, C., "Computing integrals involving the matrix exponential," *IEEE Trans. Automatic Control*, vol. 23, pp. 395–404, 1978.

[90] Wang, X., Sontag, D., and Wang, F., "Unsupervised learning of disease progression models," *Proceeding KDD '14 Proceedings of the 20th ACM SIGKDD intl conference on Knowledge discovery and data mining*, vol. 4, no. 1, pp. 85–94, 2014.

[91] Wang, Y., Resnick, S. M., and Davatzilos, C., "Spatial-temporal analysis of brain MRI images using hidden markov models," in *MICCAI*, vol. 6362, 2010.

[92] Wasserman, L., "All of statistics: A concise course in statistical inference," *Springer*, 2003.

[93] Wikipedia, "`https://en.wikipedia.org/wiki/Fixed_effects_model`,"

[94] Wikipedia, "`https://en.wikipedia.org/wiki/Nyquist_Shannon_sampling_theorem/`,"

[95] Wollstein, G., Kagemann, L., Bilonick, R., and et al., "Retinal nerve fibre layer and visual function loss in glaucoma: the tipping point," *Br J Ophthalmol*, vol. 96, no. 1, pp. 47–52, 2012.

[96] Wollstein, G., Schuman, J., Price, L., and et al., "Optical coherence tomography (OCT) macular and peripapillary retinal nerve fiber layer measurements and automated visual fields," *Am J Ophthalmol*, vol. 138, no. 2, pp. 218–25, 2004.

[97] Wollstein, G., Schuman, J., Price, L., and et al., "Optical coherence tomography longitudinal evaluation of retinal nerve fiber layer thickness in glaucoma," *Arch Ophthalmol*, vol. 123, no. 4, pp. 464–70, 2005.

[98] Wollstein, G., I. H., J., W., and et al., "Comparison of three optical coherence tomography scanning areas for detection of glaucomatous damage," *Am J Ophthalmol*, vol. 139, no. 1, pp. 39–43, 2005.

[99] World Health Organization, "A global brief on hypertension: silent killer, global public health crisis: World health day 2013," 2013.

[100] Yin, J. and Yang, Q., "Integrating hidden markov models and spectral analysis for sensory time series clustering," in *IEEE International Conference on Data Mining*, 2005.